2000-06

Authentication in SAAM routers

Szczepankiewicz, Peter J.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

AUTHENTICATION IN SAAM ROUTERS

by

Peter J. Szczepankiewicz
Luis E. Velazquez

June 2000

Thesis Advisor: Geoffrey Xie
Associate Advisor: Rex Buddenberg

Approved for public release: distribution is unlimited.

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE June 2000	3. REPORT TYPE AND DATES COVERED Master's Thesis
TITLE AND SUBTITLE : Authentication in SAAM Routers		5. FUNDING NUMBERS
6. AUTHOR(S) Peter J. Szczepankiewicz, Luis E. Velazquez		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A		10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution is unlimited.	

13. ABSTRACT *(maximum 200 words)*

Authentication is particularly important in the SAAM system because SAAM uses mobile codes, called resident agents. These resident agents are loaded onto SAAM routers dynamically, and execute on the destination SAAM router. Mobile code in the SAAM system requires an authentication scheme to prevent an outsider from sending a malicious resident agent. The primary focus of this research is to find the best-fit authentication scheme for the SAAM system.

SAAM with authentication can be used as the technical network infrastructure to support Network Centric Warfare (NCW) as described in JV2010. The prototype in this thesis authenticates new nodes that join a SAAM network using Kerberos. Signaling data, called control traffic, is authenticated with a dynamic signature key that changes every two minutes. Once a SAAM node is authenticated, its identity is protected throughout the battle.

14. SUBJECT TERMS Authentication, Encryption, Routing, Java, Key Distribution, Kerberos, Secure Time Synchronization	15. NUMBER OF PAGES 225
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited.

AUTHENTICATION IN SAAM ROUTERS

Submitted in partial fulfillment of the requirements for the degrees of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

Peter J. Szczepankiewicz
Lieutenant, United States Navy
B.A., Boston University, 1994

MASTER OF SCIENCE IN COMPUTER SCIENCE

Luis E. Velazquez
Captain, United States Marine Corps
B.S. Jacksonville University , 1992

from the

NAVAL POSTGRADUATE SCHOOL
June 2000

Authors: [redacted]
Peter J. Szczepankiewicz, Luis E. Velazquez

Approved by: [redacted]
Geoffrey Xie, Thesis Advisor

[redacted]
Rex Buddenberg, Associate Advisor

[redacted]
Dr. Dan Boger, Chairman
Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Authentication is particularly important in the SAAM system because SAAM uses mobile code. These resident agents are loaded onto SAAM routers dynamically, and execute on the destination SAAM router. Mobile code in the SAAM system requires an authentication scheme to prevent an outsider from sending a malicious resident agent. The primary focus of this research is to find the best-fit authentication scheme for the SAAM system.

SAAM with authentication can be used as the technical network infrastructure to support Network Centric Warfare (NCW) as described in JV2010. The prototype in this thesis authenticates new nodes that join a SAAM network using Kerberos. Signaling data, also called control traffic, is authenticated with a dynamic signature key that changes every two minutes. Once a SAAM node is authenticated, its identity is protected throughout the battle.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

We would like to thank Dr. Geoffrey Xie for his constant willingness to advise us with our research. His guidance and enthusiasm helped to carry this project to completion. We would also like to thank Cary Colwell for his assistance with our implementation. For math, science, our literacy, our country, and our families, we thank God.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. NCW NETWORK

Joint Vision 2010 provides the long-term vision for the US military. Network Centric Warfare (NCW) is one goal of JV2010, given the present Revolution in Military Affairs (RMA) in our environment. The RMA is caused by the explosion of Internet technology. Network Centric Warfare is the idea that several military platforms will be on-line with each other, exchanging situational awareness in real time. Sensors, decision support systems and weapons will be distributed across multiple platforms rather than being autonomously placed on single platforms. For example, an airplane in flight will accept an enemy sighting directly from a ground tank. One major assumption of JV2010 is that the underlying network technology will accommodate NCW.

NCW is a concept. The applications of NCW can run on an active network. Server and Agent Based Active network Management (SAAM) is such an active network in prototype at the Naval Postgraduate School. SAAM can be used as the technical network infrastructure component to support NCW. SAAM is a technology for the generation after next. In Internet years, it's about three years away from market.

JV2010 contains several strategic principles that SAAM addresses from a technical perspective.

- Fusion of network intelligence with platform sensors is what SAAM was designed to do. With SAAM, an airborne radar sensor can automatically open a high priority flow to a ground-based guided missile that is on the other side of the globe.

- SAAM provides information superiority via dynamically deployable resident agents, which parse information for the decision maker.
- Dynamic changes in the warfare environment may best be accommodated by a network that dynamically changes during conflict, which SAAM does. An agile organization is best supported by an agile network infrastructure. SAAM can mutate on the fly, as needed.

The following is a list of technical problems in making the NCW network and solutions that SAAM provides.

1. Problem

a. The current Quality of Service (QoS) protocols are inadequate for battle, because the network will behave like the current best effort service model. DiffServ and MPLS are some example protocols in use. The QoS model they use is based on class of service. Packets can be grouped into classes of service. The Olympic model of bronze, silver, and gold demonstrates three different classes of service. A node heading into battle will try to send important information as gold service. For example, a force recon marine who discovers an enemy platoon would want to communicate that information to friendly forces as fast as possible. With everybody sending gold traffic, QoS will fail because all traffic that is not dropped is gold. The network is now delivering best effort service, like the Internet of the 1990's.

b. QoS policy must be set ahead of time.

c. Nodes can be administratively restricted to a certain level of service ahead of time, in the interest of protecting QoS. However, this command decision would be a mistake. The theory of NCW is to allow the important information to arrive when

and where it is needed, in real time. The network must be a tool to that end, and the restriction to not allow real time traffic is in conflict with NCW.

d. Resource Reservation Protocol (RSVP) has a scalability problem. The QoS decisions are made on each router. RSVP introduces state into the routing fabric. State was taken out of IP to keep the IP protocol as simple as possible, and RSVP adds state back in. End to end state-full routing is difficult to scale.

Solution:

a. SAAM implements a better-fit QoS model for battle because the network will not fall back to best effort service. SAAM supports the flow based service model, meaning that a conversation on the network can be given its own priority if required. SAAM also supports the other two service models, best effort and class based. For clarification of these terms, please see the Spectrum of QoS Models section in this thesis.

b. SAAM QoS policies are set in real time, not only ahead of time.

c. There will always be QoS in flow-based routing, even during times of stress. Flow-based routing is a specific QoS guarantee provided on a per conversation basis. Each connection that has subscribed to flow-based routing is assigned its own flow identifier. For example, the highest priority can be granted to the CINC outgoing hotline.

d. SAAM addresses the scalability problem by using servers in a hierarchy, much like the current Internet uses Domain Name Service (DNS). RSVP can still be used in SAAM as a signaling vehicle for flow-based QoS.

2. Problem

Service level agreements are not extended across ISP boundaries with current QoS protocols. QoS is only experienced when an organization owns the entire network today. It is more likely that the military will be using civilian links for some traffic, including satellite downlinks from Globalstar for example. The NCW network must guarantee QoS from one end node all the way to the other end.

Solution: SAAM accomplishes painless interoperability between ISP's. A hierarchy of servers makes QoS decisions. One parent server between two ISP's recommends QoS levels down to two child servers. Each ISP can accept or reject this recommendation, based on internal ISP policy. Each ISP must be SAAM enabled for this situation. In addition, SAAM allows for backward compatibility with legacy QoS protocols, such as RSVP. An RSVP packet that enters the SAAM fabric will be assigned a SAAM flow ID inside the SAAM fabric, and sent back out the destination end as an RSVP packet again. QoS across ISP's is provided for by SAAM.

3. Problem

Router upgrades during battle are difficult to make today. Feature sets on routers are locked down today, restricting a QoS network from healing itself while still running packets. For example, most router operating systems usually require human intervention to add a new protocol and reboot the router to activate that protocol. A small tactical network may require a sudden boost in performance, during an ambush for example. The current Internet is not active. One minute of downtime for a reboot is not acceptable in NCW.

Solution: New protocols are added to SAAM on the fly, as needed by the flows. SAAM is an active network. Smart agents, called Resident Agents, are added to routers to do whatever is needed, including probing malicious nodes. The SAAM network can heal itself in milliseconds, even before a human is aware of technical problems.

Auto configuration is built into SAAM. The SAAM network periodically checks all nodes. This signaling traffic introduces minimal overhead. Nodes that are down are dropped out of the network, and flows are automatically rerouted around the problem nodes, all without the end user's involvement.

4. Problem

The current Internet has many security holes. One specific problem is to authenticate friendly nodes on the network. In NCW, the aircraft that flies into the battle space must be identified as friendly in real time. Today's military uses Identification Friend or Foe (IFF) traffic on separate out of band networks, and one of the greatest problems is interoperability between stovepipe IFF systems.

Solution: SAAM authentication will accommodate IFF traffic in band. The problems of interoperability are solved by virtue of all traffic using an IPv6 network. The SAAM prototype in this thesis authenticates new nodes that are added to the SAAM prototype. Initial key distribution is only discussed because the technology for this problem still does not yet exist. However, this thesis does authenticate new nodes that join a SAAM network using Kerberos and allowing for authentication in IPsec. Each signaling flow, called control traffic in SAAM, is authenticated with a dynamic signature key that changes every two minutes. This two-minute window allows the key to be very small. Signing and authentication is much faster than would be with a large long-term

key. In summary, once a SAAM node is authenticated, its identity is protected throughout the battle. A stolen SAAM node from a downed aircraft is useless to the enemy. Future smart agents could run genetic algorithms to fingerprint and thwart enemy IW attempts. This thesis supports IFF traffic in NCW.

B. RECOMMENDATION

SAAM is one possible solution to the technical problems posed by NCW, and merits consideration as the network system to support NCW.

II. DOD REQUIREMENTS FOR QOS

In addition to NCW, SAAM is also relevant to the US Navy's present-day Fleet Network Operations Centers (NOC's). The Navy has plenty of reasons to implement QoS control systems, such as the need for prioritizing unofficial e-mail packets below official Navy message packets. The ship to shore connectivity is often a bottleneck. The allocation of bandwidth on the ship to shore links is a particular QoS challenge to the Navy. Sensor and weapons systems will need real time traffic during battle. These are only a few of the Naval needs for QoS.

A. SPECTRUM OF QOS MODELS

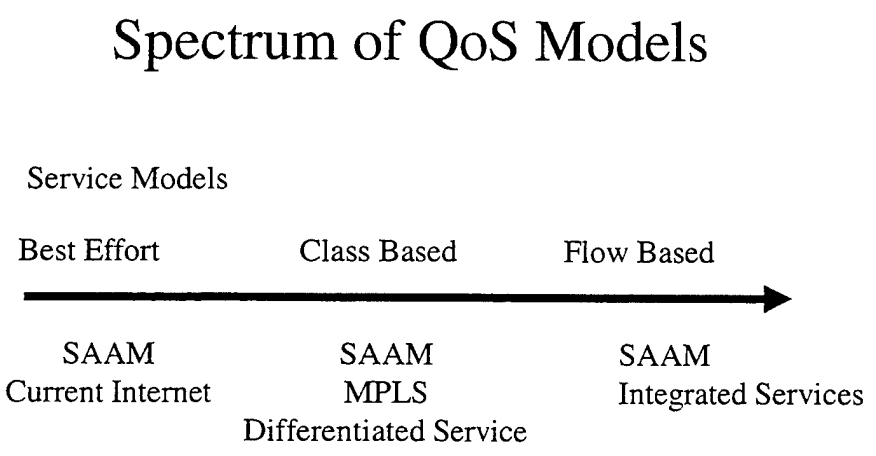


Figure 1. Spectrum of QoS Service Models

Though there are many QoS implementations, there are three basic service models. Notice that SAAM covers the entire spectrum. This figure focuses on service

models only. The figure does not show signaling protocols, such as RSVP, or packet scheduling algorithms, such as Weighted Fair Queuing, Early Random Detection, and Weighted Early Random Detection. Using signaling protocols and scheduling algorithms implement Service models.

Best effort is in effect on the Internet today. All packets on the network are in equal competition with each other. The next model, class based, groups packets into different classes. The Olympic model of gold, silver and bronze is an example of how the class based model works. Packets are grouped into one of three different classes, gold, silver or bronze. A silver packet will be delivered before a bronze packet, and a gold packet will be delivered before any other class. Within one class, all packets compete with each other. The problem with class based QoS is Service Level Agreements (SLA's) between peer networks. Peering arrangements are between Internet Service Providers (ISP's). There is no technical way to keep one ISP from sending all traffic in the gold class. Eventually, all ISP's send gold traffic and the situation is forced back to the best effort model. For this reason, someone or something has to make decisions to authorize bandwidth use.

Given the failure of human beings to effectively throttle back on low priority traffic, the concept of bandwidth broker was developed. A bandwidth broker is a special computer in the Internet that will decide which packets are allowed through at what priority. Developing a bandwidth broker is a field of research in itself and much information is available today on the World Wide Web.

The flow-based model addresses the technical problem of how to assign priority. Each conversation on the network is assigned a unique flow, similar to a tcp stream.

Each flow is assigned a priority. In SAAM, this is a function carried out inside the server. The idea of a bandwidth broker is designed into SAAM from the start. For an in depth study on how SAAM compares to other QoS schemes, please refer to [Quek].

It is interesting to note how similar the message precedence in the AUTODIN system is to the class of service protocols today, such as DiffServ. The class of service model is like ranking a message with the Olympic model. The classes are gold, silver, or bronze. In the same way, AUTODIN message precedence has been labeled as Flash, Immediate, Priority, and Routine.

	DiffServ Olympic Model	AUTODIN Priorities	AUTODIN Required Delivery Times
High	Gold	Flash	10 min.
Medium	Silver	Immediate	2 hr.
Low	Bronze	Priority	12 hr.
		Routine	24 hr.

Table 1. Comparison of DiffServ With Autodin Message Precedence

1. SAAM and Integrated Service

SAAM is based on a flow-based model, providing the guarantee that the network will never behave like a best effort network. IntServ is another QoS scheme that provides flow based QoS. The advantage of SAAM over IntServ is that SAAM offloads the flow setup calculations from the routers. IntServ relies on each router to calculate the flow next hop, whereas is SAAM the server performs this calculation only one time. SAAM conserves CPU cycles over IntServ.

B. WHY SAAM MAKES SENSE TO THE MILITARY

1. Relevance to Fleet NOC's

The Navy Fleet Network Operations Centers (NOC's) have a need for QoS. For example, the AUTODIN message system, which has been operational since the mid 1950's, has become crowded with message traffic in areas and times of crisis. Wherever the Navy was active in the world, that was exactly where our messaging system was bombarded with traffic. To address this problem, the MINIMIZE system was instituted. All Commanding Officers (CO's) were expected to review outgoing message traffic headed into the crisis area. The theory was that the CO's would participate in keeping the network open for important messages by reducing the quantity of low priority traffic. In practice, however, the MINIMIZE keyword often became an invitation for large intelligence reports deluging the AUTODIN network with flash precedence messages. The outcome of MINIMIZE was often precisely opposite of the intent. Human behavior is one limiting factor to QoS systems. Given the opportunity to decide if their traffic is important, people will tend to view their traffic as important and send the message into an already crowded network.

C. OUR WORK ON SAAM

SAAM is a software system designed to meet the needs of the Next Generation Internet, using IP version 6. A working prototype has been in operation at the Naval Post Graduate School since September 1999. [Vrable and Yarger] Several students have conducted research to enhance the prototype. Our work is to incorporate security.

1. Project Goal

The goal of this project is to develop a security system into SAAM. Our work begins where similar research has already been done but was not integrated into SAAM because the SAAM prototype did not yet exist. [Hensley and Ludden]

2. Scope of This Thesis

The SAAM project requires an authentication mechanism specifically designed to prevent unauthorized requests throughout the SAAM Enterprise. At the same time, overhead must be kept to a minimum. Initial key distribution is a problem that will be explored, along with how to prevent replay attacks, which are becoming easier to do for any Internet user. SAAM signaling traffic is vulnerable to spoofing. Because SAAM is an active network, mobile code is sent across the network. These resident agents must come from a trusted source before they are executed. SAAM authentication will allow future SAAM server and router application developers the opportunity to continue inter-enterprise communications.

We would also like to include some of our work on the business side of SAAM. The organizational behavior effects of SAAM on a tactical Information Warfare (IW) command are explored in depth. Please see the appendix, entitled "Organizational Behavior Effects of SAAM."

It is assumed that the reader has some knowledge of basic network security. If not, then please refer to the appendix entitled "Authentication Primer" or use any of the sources listed in the References.

THIS PAGE INTENTIONALLY LEFT BLANK

III. SAAM INTERNALS

A. MAJOR COMPONENTS

SAAM is an acronym for Server and Agent based Active Management. Several important ideas are contained in this name.

The SAAM system is server based by design, which makes SAAM unique among all QoS over IP systems. Some QoS systems force the overhead calculations into each router, while SAAM offloads these calculations onto a dedicated server. Thus the routers are relieved of the overhead.

Agent based means that there are smart agents in the network. These agents perform functions such as probing a node to test if the new node is an unauthorized node or is damaged. Artificial intelligence (AI) can be added to enhance these modular agents, to support backward chaining, make a decision and take action, all before human beings are involved. The entire decision cycle can take place within microseconds, near real time.

Active networking is an emerging technology in the year 2000. One unique characteristic of active networking is the ability to run applications within the routing fabric. New applications are loaded and unloaded as needed. These resident agents include core routing modules such as a routing algorithm or a queue scheduler. The mobile code used in SAAM must be authenticated to protect the network from intruders.

From the technical industry perspective, SAAM has been called several things, including layer-4 routing, active networking, QoS over IP, server based routing, state-full routing, and a smart network. SAAM incorporates all of these aspects into one system.

The current SAAM prototype uses IP version 6. IPv6 is being fine-tuned by the IETF to address the problems that IP version 4 has. All the benefits that are built into IPv6 are incorporated into SAAM by default.

1. SAAM Servers

A SAAM Server is similar to a helicopter viewing an area of highway traffic. With this view, the server is able to direct traffic to the best path between a source and destination. The best path may be the least congested path among all possible paths, or one that will optimize the sum of all resources available. The Path Information Base (PIB) module in a SAAM server performs the functions of a bandwidth broker. Flows are assigned by the PIB, allocating bandwidth that is requested. Priority is assigned via flow ID's.

2. SAAM Routers

SAAM Routers are similar to local traffic controllers. Best effort traffic may enter the Information Superhighway without any QoS guarantee. Traffic that requires a performance guarantee will have to be admitted by the routers. The routers look to the server for approval.

B. AUTHENTICITY REQUIREMENTS

In the context of communications across a SAAM network, the following attacks can be identified:

1. Disclosure

Disclosure is the release of message contents to any person or process that does not have the appropriate cryptographic key. An adversary could potentially re-route all traffic to a node where they have root access.

2. Traffic analysis

Traffic analysis is the discovery of the pattern of traffic between parties. Protection from traffic analysis is not addressed by the security in this thesis.

3. Spoofing

Spoofing is the insertion of a message into the network by a fraudulent source. SAAM is particularly vulnerable to a spoofing attack because of the use of mobile code. The security protocol for SAAM should prevent spoof attacks.

4. Content modification

Changing the contents of a message, including insertion, deletion, transposition and modification of bits and characters, is content modification.

5. Sequence modification

Sequence modification includes any modification to a sequence of messages between parties, including insertion, deletion and reordering.

6. Timing modification

Timing modification includes delay or replay of messages. One technique in common use on the Internet today is to overcome security systems by capturing the bit stream, storing it for later use and then replaying the bit stream to gain access. SAAM should have an anti-replay function in the security system.

7. Non-Repudiation

Non-repudiation is the ability to prove that a person has made a transaction. Repudiation is not an issue in SAAM security because this security concerns machines and software only.

8. Scalability

SAAM is a scalable system, using a hierarchy of servers that is similar to domain name system (DNS). Security should not violate scalability.

C. PROBLEMS WITH SECURITY

Consider two inherent problems with security, brittleness and scalability.

1. Brittleness

All nodes can be secured with one global key that is the same throughout the enterprise. However, a compromise anywhere is a compromise to the whole enterprise.

2. Scalability

All nodes can be secured by using pairs of keys for each session. This overcomes brittleness because the compromise of any one node would not harm the rest of the system. An exception to this is the central KDC in Kerberos, but the KDC is assumed to be physically secure. The problem with using session key pairs is scalability as described below.

Any security scheme that uses a shared secret assigns a session key for each connection in a network. Each pair of nodes shares a pair of symmetric keys. As demonstrated in the graph below, the number of sessions that can occur in a 4-node network is only six sessions. That is, six key-pairs must be present in this network.

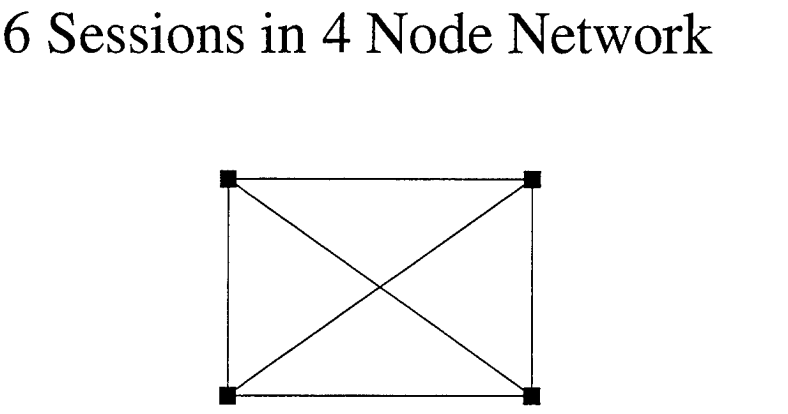


Figure 2. Six Sessions Are Needed For a Four-Node Network

Given 5 nodes, there must be 10 sessions. Given 6 nodes, there must be 15 sessions. The mathematical relationship between the number of nodes and the number of sessions is as follows:

N = number of nodes

S = number of sessions

$S = (N * (N-1)) / 2$

As the number of nodes increases linearly, the number of sessions increases at a much faster rate.

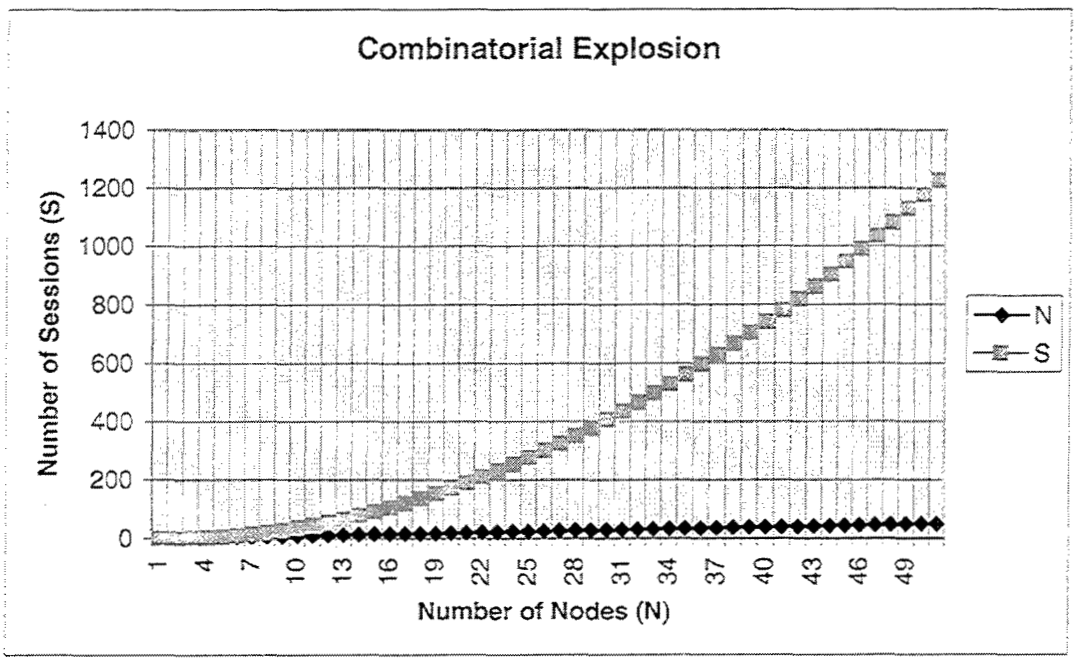


Figure 3. Combinatorial Explosion

The problem with combinatorial explosion is the management of so many keys. Given that the maximum number of nodes on one SAAM active site is 40, there will be 780 sessions.

D. REASONS FOR KERBEROS

Kerberos is used during phase one of the protocol, which establishes machine-to-machine authentication only. Kerberos was developed to protect remote logins from students across an untrusted network. SAAM routers will be logging into the SAAM Active Site over an untrusted network. We chose Kerberos because it was ideally suited to protection from eavesdropping on the network. SAAM requires authenticity but not confidentiality to avoid overhead.

How we chose Kerberos is interesting to note. We began with a simple matrix to compare several security schemes against each other. A decision support software tool was then used to try to quantify the utility of each authentication system, and to rank the best alternatives.

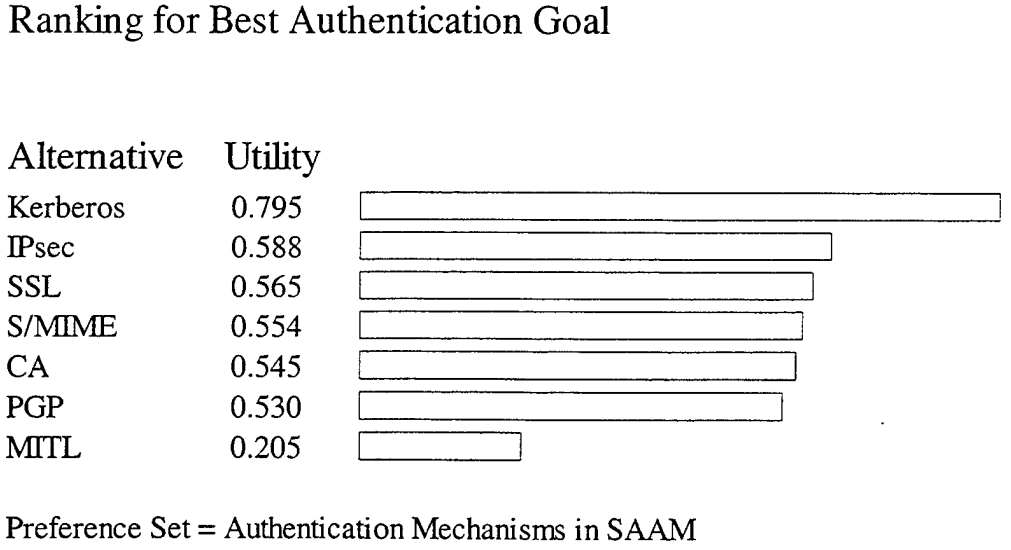


Figure 4. LDW result - Ranking for Best Authentication

Finally, we discovered that the authors of rfc2747 were faced with the same decision to find an authentication scheme for a distributed protocol. They also chose to

use Kerberos. For more information on how we chose Kerberos, please refer to the appendix entitled "Selecting and Authentication System."

E. OUR SETUP

Our solution avoids both the brittleness and scalability problems. The authenticity protocol is comprised of two phases. Phase one of SAAM security will use Kerberos. Kerberos will establish a trusted session between two nodes, using a Trusted Session Key pair. The use of session key pairs avoids the brittleness problem, but does have the scalability problem.

To overcome this scalability problem, SAAM will use a global key that is valid for a very short period of time. This global key is called the SAAM recognition key. In that way, Trusted Session Key pairs are kept to a minimum. Only a single recognition key is active at any one time, independent of the number of nodes.

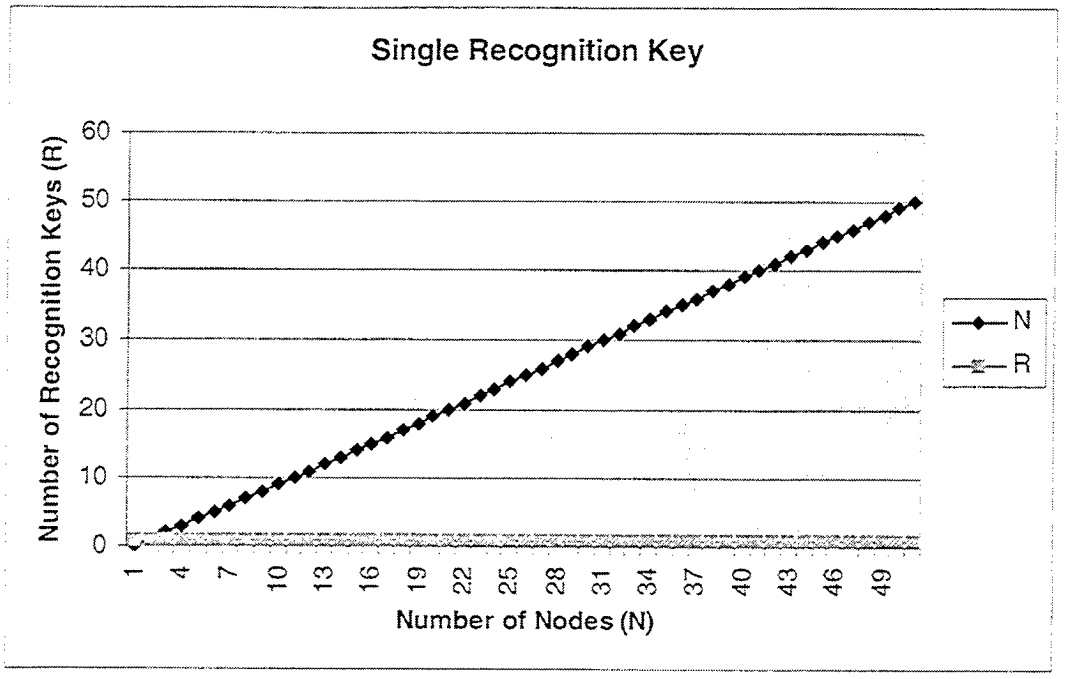


Figure 5. One Global Recognition Key Independent of Number of Nodes

The global key does, however, have the brittleness problem. The brittleness problem is addressed by periodically refreshing the global key. A table of keys will be distributed periodically, and all routers will authenticate with each other by stepping through the table of keys.

The timing of this step will be an important issue to solve. The length of time that one key can remain secure must be less than the length of time that the key could be brute force cracked. In the case of a one-way hash algorithm, such as MD5, the key shift should occur every two minutes. Network timing protocol (NTP) is accurate to milliseconds and may be the best way to solve the timing-jitter problem. How to keep this time in synchronization will be explored.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. DESIGN

A. CHAPTER SUMMARY

This design chapter is organized into five major parts, beginning with the introduction. The major assumptions portion explains what assumptions were made to make this protocol a reality. The functions of the authentication protocol are detailed in this chapter. Three different scenarios are identified where SAAM will need to use the authentication system, and these scenarios are described in detail, using timing diagrams. The interactions between the classes are described, using UML-like diagrams. Finally diagrams of packet structures are provided.

1. Definitions

The following definitions describe the state of a node:

New Node – A new node is untrusted and unrecognized. A new node is different from an intruder because a new node shares a node secret with the KDC.

Trusted – A node in the trusted state has a Trusted Session Key with a node that is recognized in place. A trusted node has a session with an already recognized node. A trusted node is not necessarily recognized.

Recognized – A recognized node will have the SAAM key table in place and actively working. The Recognition Key is active in a recognized node.

Invalid – A node that has no node secret. For example, a node that has been captured by the enemy is removed from the SAAM network by revoking the node secret from the KDC.

The following definitions describe the key types.

Node Secret - K_i This is the long-term key that is in place before any traffic runs over the net. In Kerberos terminology, the node secret is called the secret key. This node secret is shared between the KDC and node i.

Trusted Session Key – $\hat{k}_{lm}$ This is the final Kerberos session key shared between node i and node j. In Kerberos terminology, the Trusted Session Key is called the session key. Each realm will have a maximum of 40 Trusted Session Key pairs.

Recognition Key – k [ ] This is the active SAAM session key inside the Recognition Key Table throughout the SAAM active site.

The following definitions describe the message types with Java class names.

JoinRequest.java – A message sent when a new node wants to gain the Trusted Session Key. The JoinRequest message is used before Kerberos begins. This message goes from an untrusted New Node to a trusted neighbor.

TrustedSessionRequest.java – This message goes from a Recognized Node to the KDC. This class is an abstraction of all the messages that the Kerberos protocol sends in order to gain the Trusted Session Key. The TrustedSessionRequest message is used to begin the actual Kerberos traffic.

TrustedSessionResponse.java – This message goes from the KDC to a Recognized Node. Again, this class is an abstraction of the messages within the Kerberos protocol. It is important to note that the TrustedSessionReponse message does not show the full Kerberos Trusted Session Key distribution. In Kerberos, the ticket is relayed to the target host, along with an authenticator. Mutual authentication occurs when another authenticator is sent from the target to the trusted neighbor. Kerberos provides some

security against reply attacks by using a time stamp within the authenticator. TrustedSessionResponse is hiding all of this traffic from SAAM.

KeyTableRequest.java – A message sent to gain the Recognition Key Table. This message goes from a trusted and unrecognized node to a Recognized parent.

KeyTableResponse.java – A message that contains the Recognition Key Table encrypted with the Trusted Session Key. This message goes from a Recognized parent node to a child node that has just become recognized.

TimeResponse.java – A message that synchronizes the time from parent to child. This message goes from a recognized node to another recognized node.

B. MAJOR ASSUMPTIONS

1. First Key Already in Place

The very first long-term key, called the Node Secret in this authentication protocol, is crucial to the subsequent distribution of all other session keys. The first key must already be in place before the authentication protocol begins. No matter what mechanism is chosen, interaction from a human being will be required at some point. Some possible implementations of first key distribution include PKINIT, PKI, SecureID card, Smart card, Certificate Authority, and hardcoded secret from factory in EPROM. For the purposes of this thesis, it is assumed that the node secret is hard coded from the factory in an EPROM chip.

2. Time Sync on Kerberized Hosts

It is assumed that the BIOS clocks on the SAAM routers are reasonably close to each other before Kerberos sends authenticator messages, which are based on time. The time sync messages in the authentication protocol are sent after Kerberos, and just after a node becomes recognized.

3. KDC is Physically Secure

The Kerberos Key Distribution Center must be secure in order for Kerberos to function. It is recommended that all unneeded tcp and udp ports be blocked to and from the KDC. In addition to software controls, the KDC should be physically locked in a closet to prevent physical access.

4. Proactive Key Refresh

It is assumed that the SAAM authentication system will know when keys are about to expire and take action to refresh those keys before expiration. Kerberos does this automatically. The code for the Recognition Keys will have to include intelligence to do a proactive Recognition Key Table refresh.

5. Mutual Authentication

It is assumed that mutual authentication will always be used in this protocol. Mutual authentication means that both the kerberized hosts are authenticated to each other. For example, RouterA wants to talk to a target, RouterB. RouterA obtains a ticket from the KDC and passes the ticket plus an anti-replay authenticator to RouterB. At this

point, RouterA is authenticated to RouterB because only the trusted KDC could produce this ticket. RouterB, the target, replies to RouterA with another authenticator message encrypted with the Kerberos session key. Thus, the target is authenticated to the requestor.

C. AUTHENTICATION PROTOCOL FUNCTIONS

The authentication protocol performs the following high level functions: (1) first long term key distribution, which is called the Node Secret distribution in this document, (2) router to router authentication with Kerberos, (3) generation of Recognition Key Table, (4) encryption and distribution of Recognition Key Table, (5) secure time synchronization, and (6) authenticated SAAM signaling traffic.

1. Node Secret Distribution

The first key is already in place, as discussed in the major assumptions.

2. Machine to Machine Authentication

Kerberos is the chosen protocol for machine-to-machine authentication. In addition to authentication, Kerberos protects against replay attacks to some degree.

a. Kerberos Box Model

We used the Kerberos Box model developed by Brian Tung in his book, Kerberos. During our thesis brief, this was a standing model with written messages, cardboard boxes, locks and keys to represent encryption throughout the Kerberos

protocol. This live demonstration solidified our understanding of Kerberos. The function of tickets and authenticators was made very clear.

The basics of Kerberos are that when Alice wants to talk to Bob, Alice first requests for permission from the KDC. The KDC grants a session key for the conversation between Alice and Bob. How this session key is distributed securely in interesting. The KDC encrypts one session key with Alice's long-term key. Then the KDC encrypts the other session key with Bob's long-term key. This second message is called the ticket. The two messages are sent to Alice. Alice can decrypt the first session key with the long-term key. She cannot decrypt the ticket. Alice checks the time on her watch, writes the time in a message, and encrypts the message with the session key. This message is called the authenticator. Alice sends the ticket and authenticator to Bob. Immediately, Bob cannot open the authenticator, but he can open the ticket with his long-term key. Bob decrypts the ticket and gains access to the session key. Bob then uses the session key to open the authenticator. If the time from Alice is within a 5-minute window, then Bob trusts Alice. Bob then makes his own authenticator and sends it to Alice. Mutual authentication takes place to prevent replay attacks from Bob or Alice's packets.

The box model idea was extended to include the entire SAAM authentication protocol. Step one is Kerberos and step two involves the Recognition Key.

b. Win2K Kerberos Action Diagrams

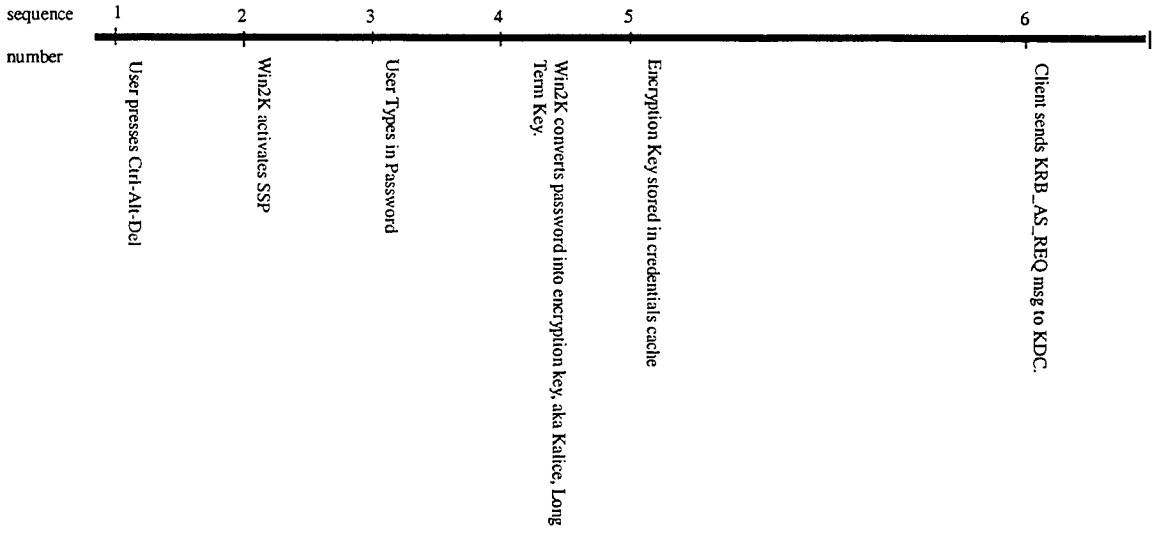
The Kerberos activity inside of Windows2000 is shown below in the form of action diagrams.

ACTION DIAGRAM

AUTHENTICATION WITH KERBEROS ON WIN2K

Msg 1 - Client to KDC
Authentication Service Exchange
Once per user logon session.
KRB_AS_REQ

KRB_AS_REQ msg parts:	
I	II
Alice, TGS	Kalice(Alice, time)
userid, svc name	Kuser (user, time)
"Alice wants TGS"	Preauthenticated data

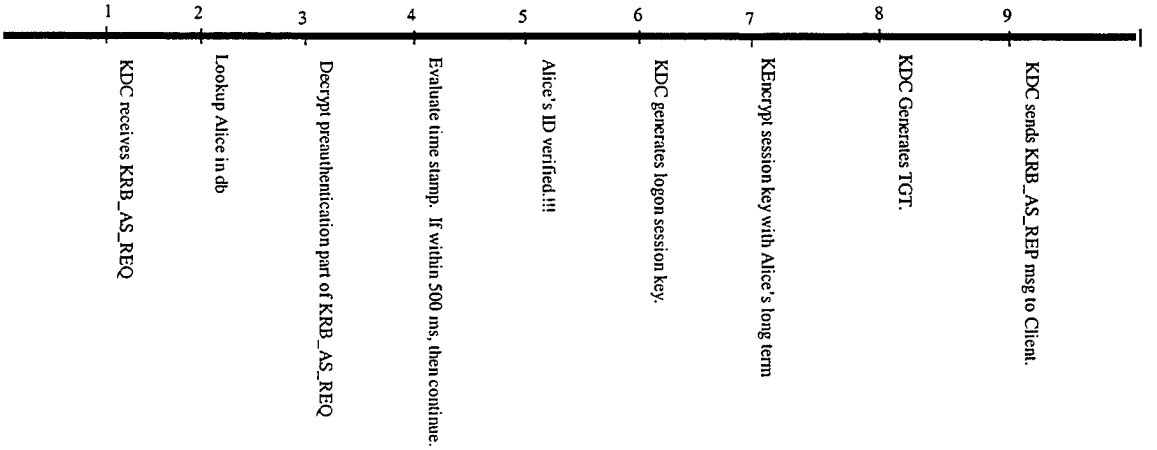


ACTION DIAGRAM

AUTHENTICATION WITH KERBEROS

Msg 2 - KDC to Client
Authentication Service Exchange
Once per user logon session.
KRB_AS_REP

KRB_AS_REP msg parts:	
I	II
Kalice	TGT
(use Salice for TGS)	Ktgs
	(use Salice for alice.)



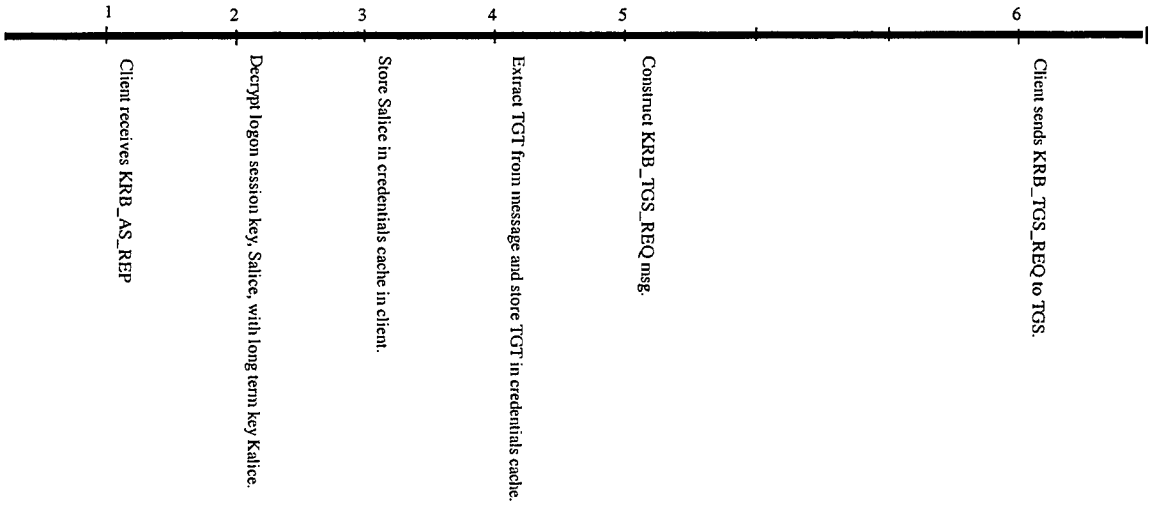
ACTION DIAGRAM

AUTHENTICATION WITH KERBEROS

Msg 3 - Client to TGS
Ticket Granting Service Exchange
Once per type of service
KRB_TGS_REQ

KRB_TGS_REQ msg parts:

I	II	III
Alice wants Bob. User Name: Alice Bob is the name of service for which user wants a ticket.	Salice{Alice,time} Salice is Alice's long term key. Authenticator is User-name, time}.	TGT Ktgs{use Salice for Alice}



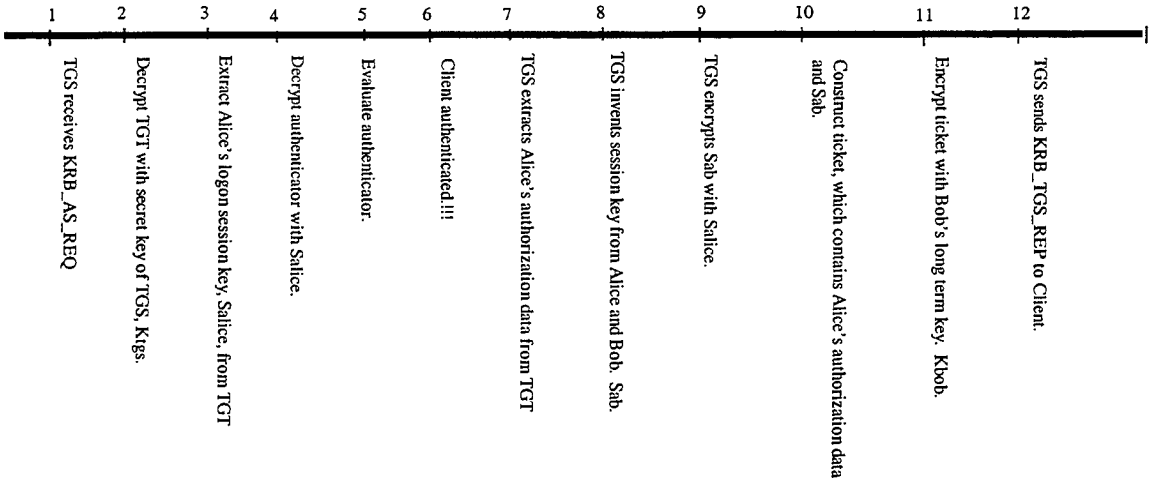
ACTION DIAGRAM

AUTHENTICATION WITH KERBEROS

Msg 4 - TGS to Client
Ticket Granting Service Exchange
Once per type of service
KRB_TGS_REP

KRB_TGS_REP msg parts:

I	II
Salice(uses Sab for Bob).	ticket Kbob (usesSab for Alice).



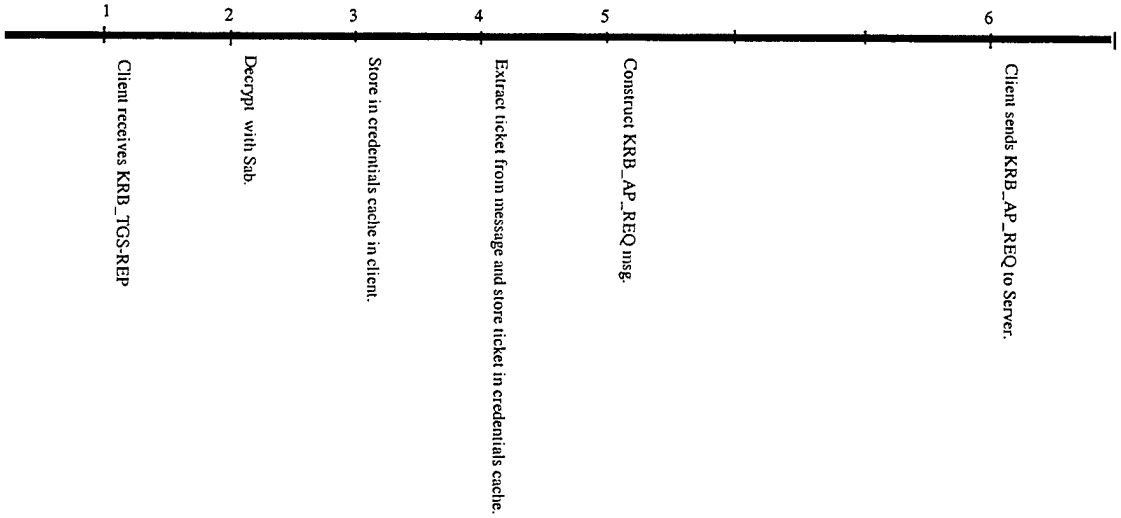
ACTION DIAGRAM

AUTHENTICATION WITH KERBEROS

Msg 5 - Client to Server
Client Server Exchange
Once per service session
KRB_TGS_REQ

KRB_AP_REQ msg parts:

I	II	III
Sab{Alice,time} Authenticator encrypted with the session key for the service.	ticket obtained from TGS Exchange.	flag Does Alice want mutual authentication from Bob?



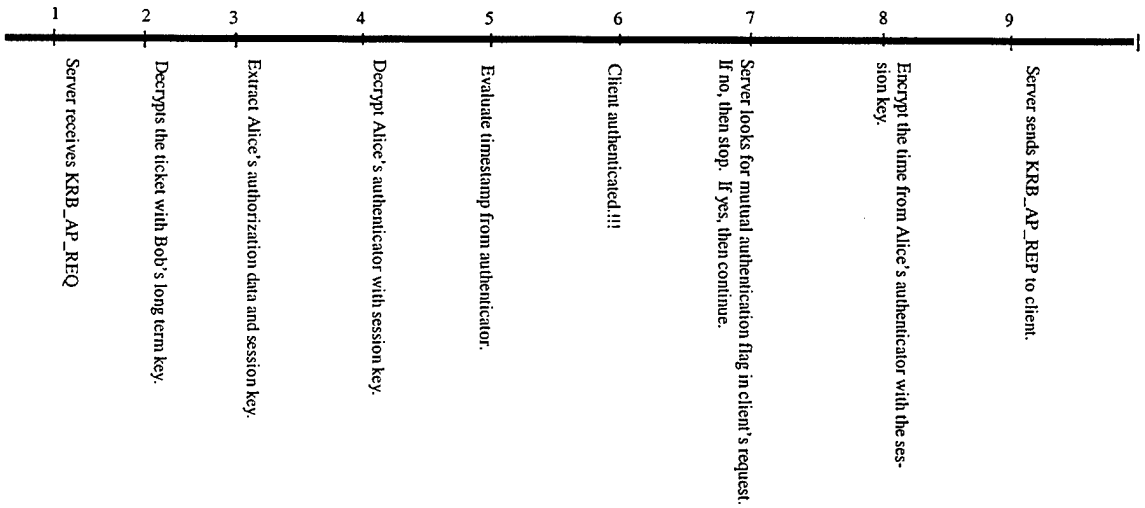
ACTION DIAGRAM

AUTHENTICATION WITH KERBEROS

Msg 6 - Server to Client
Client Server Exchange
Once per server session.
KRB_AP_REP

KRB_AP_REP msg parts:
I
Sab{time}

-note that mutual authentication is optional.



ACTION DIAGRAM

AUTHENTICATION WITH KERBEROS

Step 7 - Server and Client talk.

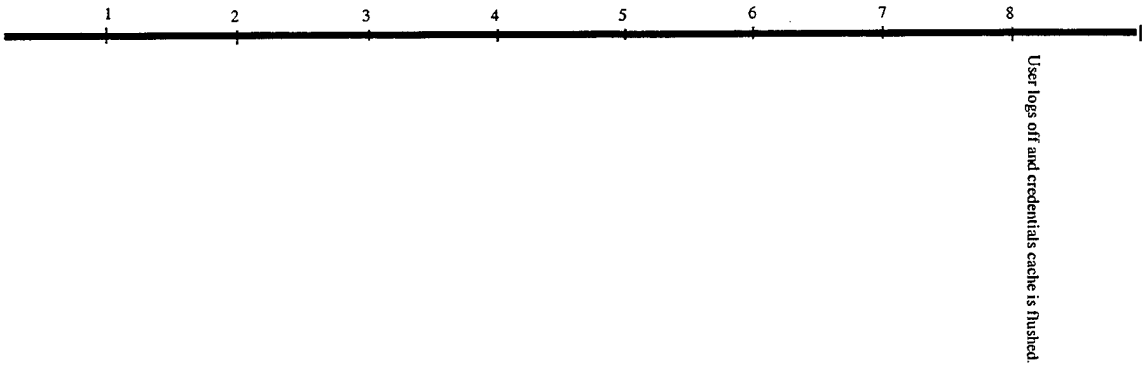


Figure 6. Action Diagram of Kerberos.

c. SSPI

Windows2000 provides an API to call Kerberos, called the Security Support Provider Interface. By using this API, applications can be Kerberized. This API requires function calls in C code.

There are five different ways that Java can call Kerberos in Win2K, as shown below. This figure and the following text is a direct quote from the Microsoft white paper, The Security Support Provider Interface (SSPI). This is an Operating Specific description, focusing on one computer. Note that it is not inherently network centric and does not describe the bits on network.

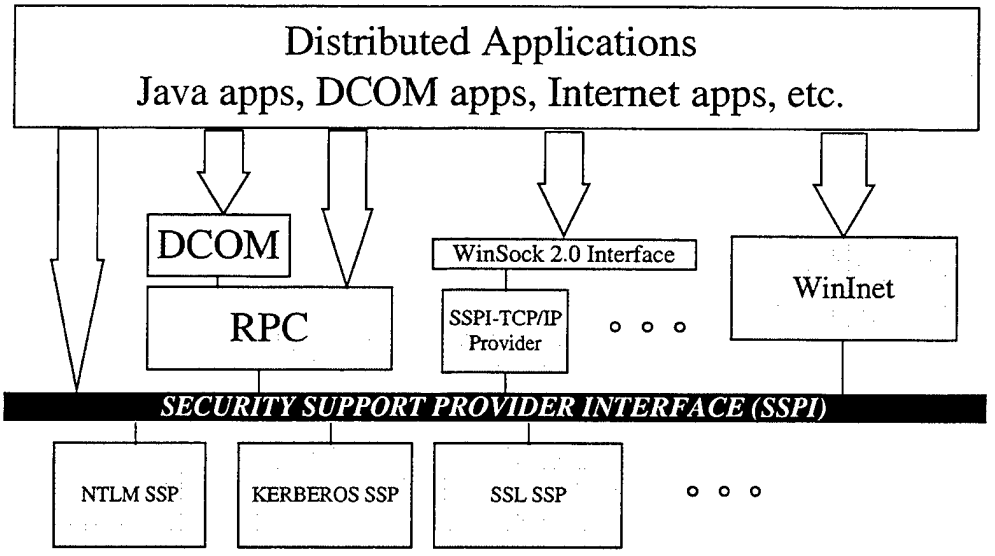


Figure 7. SSPI and the Windows NT Security Model

The figure above shows the options available to application developers for building distributed applications. The SSPI provides an abstraction layer between application-level protocols and security protocols. The following are some of the ways to use SSPI services:

- Traditional socket-based applications can call SSPI routines directly and

implement the application protocol that carries SSPI security-related data, using request and response messages.

- DCOM applications provide the best level of integrated security features. Applications can use DCOM to call security options, which are implemented using authenticated RPC and SSPI at lower levels. Applications do not call SSPI APIs directly.
- WinSock 2.0 extends the Windows Sockets interface to allow transport providers to expose security features. This approach integrates the SSPI security provider into the network stack and provides both security and transport services through a common interface.
- WinInet is an application protocol interface that is designed to support Internet security protocols, such as Secure Sockets Layer (SSL), over Internet protocols. The implementation of WinInet security support uses the SSPI interface to the Secure Channel (Windows NT implementation of SSL) security provider. From Ref. [MS White paper]

d. C code

Expertise in C code is needed to call Kerberos functions.

e. JNI

The SAAM prototype is written entirely in Java. In order for SAAM to use Kerberos, Java must call C code functions. The link between these two different languages is Java Native Interface (JNI). Expertise in JNI is needed.

f. Java

Java expertise is needed because the SAAM prototype is written in SAAM.

g. Other Sources of Kerberos

(1) MIT provides source C code for Kerberos from their web site. Since this Kerberos is also written in C, expertise in C, Java and JNI is still needed.

(2) University of Illinois provides some beta source code, which is a Java wrapper for the MIT C Kerberos. We did not try this code for this thesis, though it is important to note that this wrapper exists.

(3) JCSI is another Java version available from the following web site: http://security.dstc.edu.au/projects/java/release3.html. JCSI is another Java wrapper for MIT's Kerberos.

h. Generation of Recognition Key Table

The Recognition Key table is generated using SQL Server.

The Recognition Key table is generated while Kerberos is running. These two events can run in parallel. The Recognition Key Table is distributed upon SAAM network boot-up, and then every 24 hours at midnight.

i. Encryption and Distribution of Key Table

The Trusted Session Key is used to encrypt and Recognition Key table. There is a c function available in the SSPI called EncryptMessage() that should work. This encrypted object is then sent to destination nodes.

j. Secure Time Synchronization

Secure Time Synchronization will be carried out after a node becomes recognized. The time synchronization message is appended with the KeyTableResponse message. Time resolution to the millisecond is required, not necessarily microsecond resolution.

The external time source can be any time server, such as the US Naval Observatory time server over the Internet using NTP. Another option is to use a hardware clock attached to the primary SAAM server. External clocks are sometimes used to

prevent the time from being changed covertly. At the very least, the human eye could catch that the time has been altered. Another option for the time source is the native windows 2000 time service. There are several ways to synchronize time.

k. Authenticated SAAM Signaling Traffic

Once the recognition key tables are in place, the sender and receiver have synchronized time because of the time sync message that arrived with the recognition key table. Based on the time, the sender selects a key to sign the signaling message. Based on the time, the receiver is likewise ready to authenticate the message with the same recognition key. Recognition keys are symmetric. The sender uses the recognition key to run a hash algorithm on the message. The MD5 hash algorithm may be used for demonstration purposes. In the final deliverable protocol, a hash algorithm that is more secure than MD5 should be used. The output of the hash function is the message authentication code (MAC).

The sender then formats the packet for transmission as follows.

Key-prefix MAC Message

The key prefix field is placed in the header to optimize performance on the receiver. The receiver has the recognition key in memory, along with the associated key-prefix. The receiver begins to read the packet before the entire packet is buffered. The Key-prefix is checked before any CPU cycles are devoted to the hashing. If the key-prefix does not match the expected value, the packet is dropped. This is a fast reject technique to reduce overhead. If the key-prefix does match the expected value, then the receiver performs the hash function. If the MAC produced on the receiver matches the MAC in the packet,

then this packet is authenticated. SAAM continues to process this signaling message. If the MAC's do not match, then the message is discarded.

D. AUTHENTICATION SCENARIOS

Authentication in SAAM will happen in three distinct scenarios. They are (1) nascent network, (2) New Host joins SAAM Active Site, (3) Key Table change. The following diagrams demonstrate these three scenarios.

1. Nascent Network Scenario

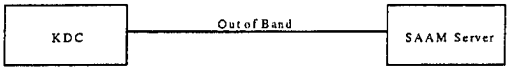


Figure 8. (a) Topology of Nascent Network Scenario

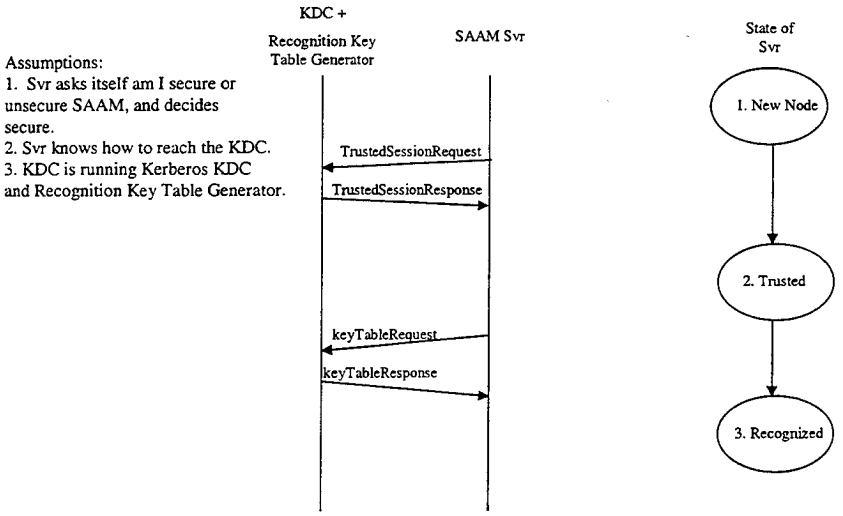


Figure 8. (b) Timing Diagram of Nascent Network Scenario

Figure 8. Nascent Network

When a new SAAM network is bootstrapped two hosts are concerned with authentication, the KDC and the first SAAM server. The ServerAgent module on the SAAM server must ask the question – will I start a secure or unsecure Active Site?

Note that we could make a security flag in the DCM message to allow the net to become secure on the fly. The net could be in an unsecure state, and move up to a secure state. That way, only the primary server has to make this security decision. The primary server could also decide to move the network down to an unsecure state in the very next DCM cycle.

The decision for secure AS is made. The second question is how will the server reach the KDC. There are several options, including (1) hard coding the IP address into the SAAM server, or submitting a host name in a configuration file that an administrator can edit before bootstrap, (2) directly connect the KDC to the server, and (3) use a traditional routing discovery protocol such as OSPF. The server is in an untrusted, New Node, state at this point.

The KDC is running two software modules. One is the Kerberos KDC, and the other is the Recognition Key table generator.

The SAAM Server sends a TrustedSessionRequest and the KDC responds with a TrustedSessionResponse. At this point, the Kerberos Trusted Session Keys are in place between the two hosts and machine-to-machine authentication is complete. The server sends a KeyTableRequest message to the Key Generator. The Key Generator encrypts the Recognition Key table using the Trusted Session Key, and sends this encrypted Recognition Key table to the server. At this point the server is in a recognized state.

2. New Host Join Scenario

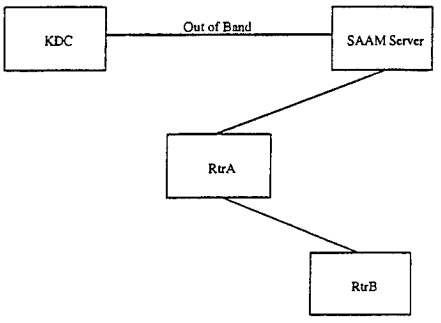


Figure 9. (a) Topology of New Node Join Scenario

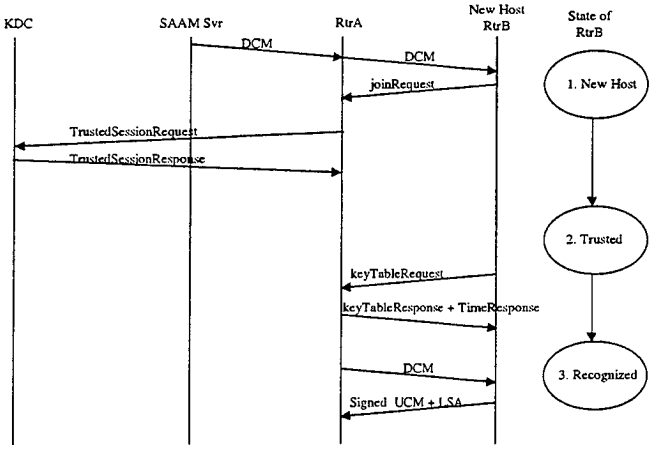


Figure 9. (b) Timing Diagram of New Host Join Scenario

Figure 9. New Host Join Scenario

A Downward control message initiates the process. A new node router platform will wait until it receives the incoming DCM message from the network. This DCM message will not be authenticated since the new node does not have the recognition Keys of the day within its volatile memory. Once the DCM has been received the new node will send a request to the neighboring unit via a JoinRequest message. This message will go to the neighboring node where the neighboring node then forwards the message along to the KDC.

It is necessary to send the JoinRequest message to the neighbor rather than straight through to the KDC because the SAAM prototype does not yet route best effort traffic. That is to say, all messages in SAAM must have an assigned flow id. Since the new node is not yet a part of SAAM, there can be no flow id assigned. The neighbor node (Router A) is already active and can send messages with an assigned flow id.

If Kerberos fails to renew it's Trusted Session Keys, the Trusted Session Key will expire. A router that has an expired Trusted Session Key will also wait until the next DCM message arrives to initiate a request to join the SAAM Realm. This will allow the same module within the Security Manager to process both cases of either a new Join Router or an expired router. In both of these cases the 720 SAAM Keys are non-existent or have expired thus demonstrating the flexibility of the SAAM Security Manager to handle both cases with the same function calls.

A Trusted session request message is sent. The neighboring SAAM Router, Router A, sends the Trusted Session Request message. The session request message is requested on behalf of the SAAM recognized neighbor and the new join router up to the

KDC. This message traffic is then is processed via normal SAAM routing, which is flow based, up to the master KDC.

The KDC generates a Trusted Session Key for the new node and requesting neighbor. The Kerberos trust mechanism queries its database of long term Node Secrets to determine if the new node request is valid. The message request is authenticated since it traveled along the SAAM Recognized flow based routing mechanism. [Akkoc] If both the requesting neighbor and the new node secret keys are found in the KDC secret key ring then the KDC will create a trusted session key that is sent back to the requesting router via the SAAM Recognized flow base routing.

The KDC sends the Trusted Session Key to the requestor, Router A, for a new node. In Kerberos, a ticket is an encrypted message that contains the session key. The requestor passes this ticket onto the target host, RouterB. The ticket is encrypted with the Node Secret Key of Router B. A Kerberos Trust is established between the requesting neighbor and the new join router.

The neighbor forwards the ticket to the New Node. Upon receipt of the messages from the KDC, the neighbor will open its message and extract the new session key that can only be utilized between the neighbor router and the new node router. At that point the neighboring router will send the message that is designated for the new node. The message format follows:

Ticket (Encrypted with New Node – Node Secret [Trusted Session Key])

The neighboring router will also send along an authenticator. The authenticator is a time stamp message encrypted with the newly created session key that only the neighbor router and the new join node can decrypt. When Router B decrypts the authenticator, the

time on Router B is checked to verify is the authenticator was created recently. If not, then it is likely that a replay attack is in effect and the session is discontinued.

The new node sends a Key Table Request to the neighbor, Router A. The new node then receives both the authenticator message and the ticket. The new node decrypts and extracts the new Trusted Session Key that is only valid between Router A and Router B.

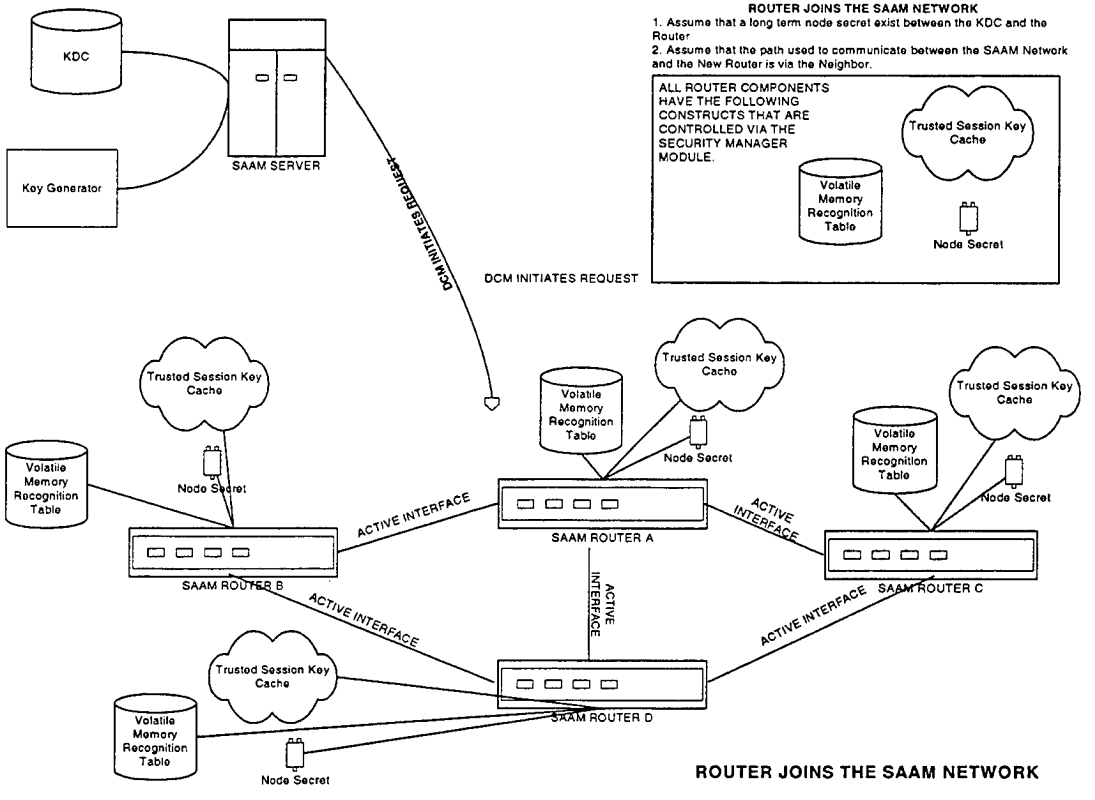


Figure 10. Router Joins the SAAM Network

The neighbor sends Recognition Keys to the New Node. Router A now has a Kerberos Trust communications link between itself and the new join router. That is why Router B is said to move to a Trusted state. The neighbor router will copy its

Recognition Keys from its volatile memory location to give to the new join router. First, the Recognition Key Table is encrypted with the Trusted Session Key.

The new node can now authenticate DCM messages. The New node now has the recognition keys in volatile memory and can process DCM and forward DCM information down its interfaces to any other routers waiting to join the SAAM realm.

3. Scenario 3. Key Table Change

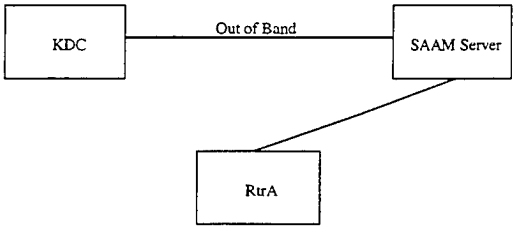


Figure 11. (a) Topology of Recognition Key Table Change Scenario

Assumptions:
1. Recognition is about to expire. This inititates at about 2300, 1 hour before expiration.
On Svr, SvrAgent knows that it is close to expiring, so the securityManager is triggered.
2. Each router is responsible to Maintain TRUST with its Neighbors.
3. Unsolicited KeyTableResponse Is sent from Svr to children.

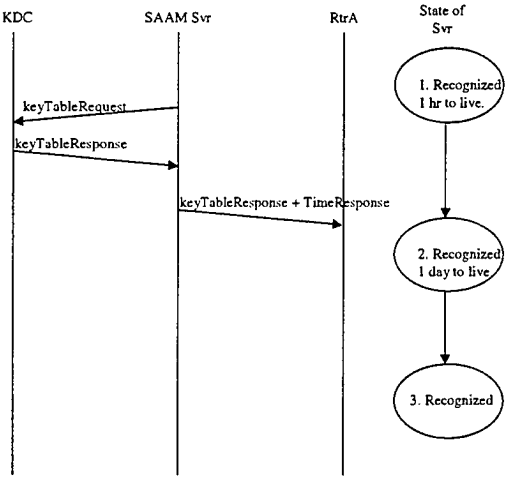


Figure 11. (b) Timing Diagram of Recognition Key Table Change Scenario

Figure 11. Key Table Change Scenario

E. JAVA CLASS FILE INTERACTIONS

The OSI model is used here to demonstrate where Kerberos will be implemented in the SAAM prototype. Keep in mind the emulated nature of the prototype as shown in Yarger and Vrable's thesis chapter 4 on modeling. Recall that the control executive and resident agents are all part of the emulated application layer. The packet factory runs at the data link layer, beneath the routing algorithm in the network layer. The emulated physical layer contains the translator. This physical layer will also contain Kerberos. Kerberos is in the physical layer because our implementation uses the Kerberos, which is built into Windows2000. Because Kerberos is in the physical layer, Kerberos can provide services to any component in the SAAM prototype, providing flexibility. The figure below shows where Kerberos fits in the SAAM prototype, using the OSI model.

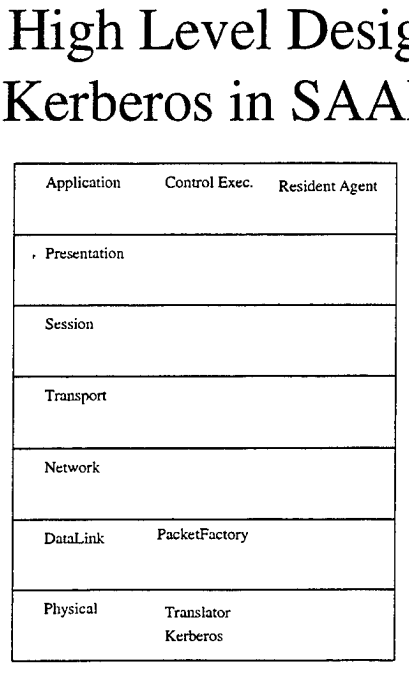


Figure 12. Emulated OSI model - SAAM Prototype with Kerberos Authentication

In very broad terms, the entire SAAM system uses the entire Kerberos system.

High Level Design 1.0

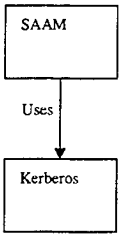


Figure 13. High Level Design - Level 1.0.

Both SAAM and Kerberos are more complex systems than the figure above shows. Within SAAM, there is the control executive class, which is involved in all control traffic, but not involved in data traffic. Regular data traffic, a.k.a. SAAM traffic, does not need authentication. Remember that overhead is a consideration when implementing authentication. SAAM is concerned with authenticating control traffic only. Control traffic is like inter-routing protocols on Cisco routers, which can change routing tables. Control traffic is also called signaling traffic.

The control executive will be kept as small as possible, similar to the principle of the micro-kernal in Microsoft's Windows NT operating system. The Packet Factory is another class that all control traffic uses. Adjustments to SAAM will be done in the Packet Factory whenever possible.

The Security Manager object will take care of all the Kerberos services for SAAM. It is interesting to note that the Security Manager is placed similar to the DiffServ bandwidth broker. The Security Manager hides the complexity of Kerberos from SAAM. The Security Manager will receive authentication requests and return a Boolean value, true or false, if the authentication passes or fails. In general, an authentication that passes will return control back to the packet factory for continued execution. A failed authentication will result in the message being dropped. For testing purposes, we will need to be notified when authentication fails with an error message. For the real SAAM deployment, no such error messages should be delivered to prevent an attacker from gaining any more knowledge about the system. The figure below shows the call from the Packet Factory to the Security Manager.

High Level Design 2.0

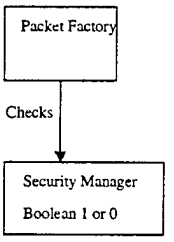


Figure 14. High Level Design - Level 2.0.

Within SAAM, there are several classes, which communicate with the control executive. They are the packet factory, the routing algorithm, the Inbound Interface, and the translator. The Security Manager will be added as shown below.

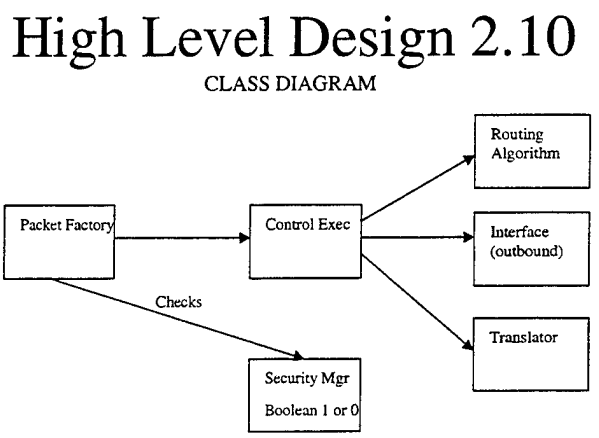


Figure 15. SAAM Classes in Contact With the Control Exec

Each of these classes and their communication with the Control Executive are discussed below.

The Packet Factory builds packets by aggregating the individual message updates into a packet and finally appending a header. The new packet is communicated to the Control Exec. The packet factory also receives inbound packets and parses the packet for messages in the payload.

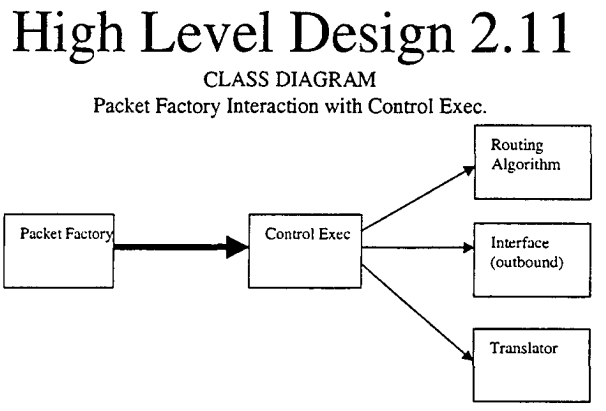


Figure 16. Packet Factory Communicates with Control Exec

The Packet Factory calls the Security Manager to do the following:

(1) Authenticate all inbound (mobile code) resident agents before execution.

(2) Authenticate outgoing code to destination hosts. Authentication of control traffic requires the Recognition Keys.

Further defense can be added to SAAM in the Security Manager. For example, a method could instantiate a virus wall (virus vacuum on a bastion host). Some work is also in progress to add a packet filter to SAAM.

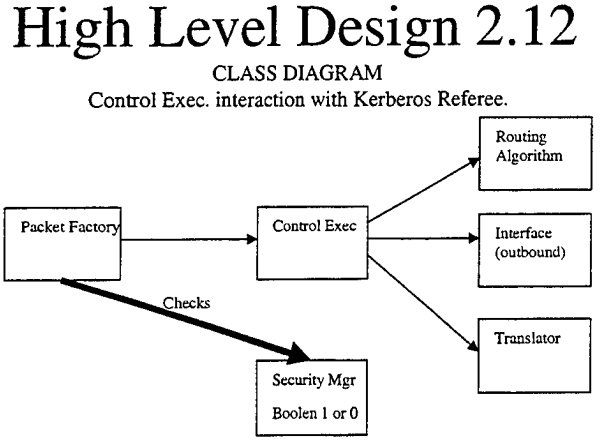


Figure 17. Packet Factory Calls the Security Manager

Control Exec sends two types of messages to Routing Algorithm:

(1) FlowRoutingTable updates

(2) ARPCache updates.

The packet which was the source of these updates must have been already authenticated. The Recognition Key has already been used by this point. If not, then send message into the bit bucket.

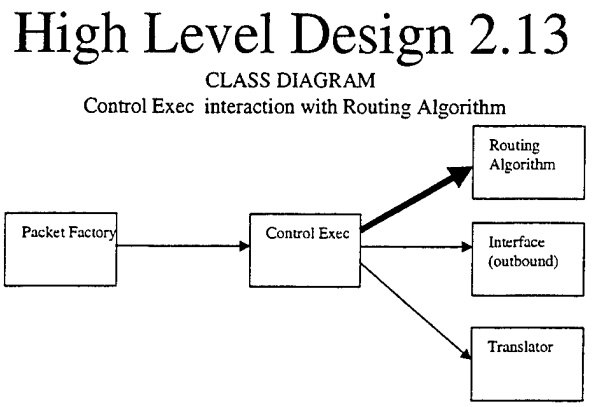


Figure 18. Control Exec Communicates With the Routing Algorithm

The Control Exec sends a new scheduler to the outbound interface. A new scheduler may help route traffic in congestion state. Again, this new scheduler arrived in the form of a resident agent, which must have already been authenticated by this point. If the message is not authenticated then the message is dropped.

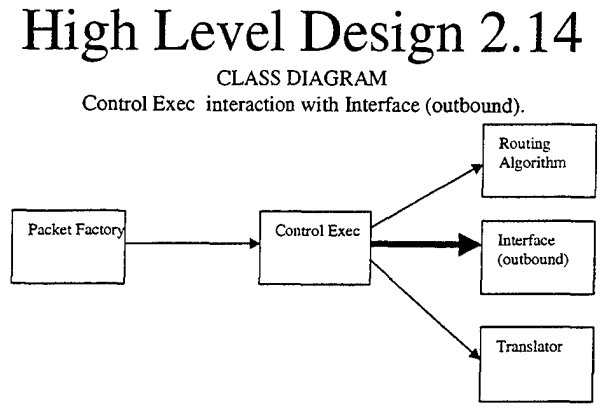


Figure 19. Control Exec Communicates With the Outbound Interface

The control exec sends emulation table update messages to the Translator. These updates have been parsed out of a packet beforehand, and so the packet must have been authenticated.

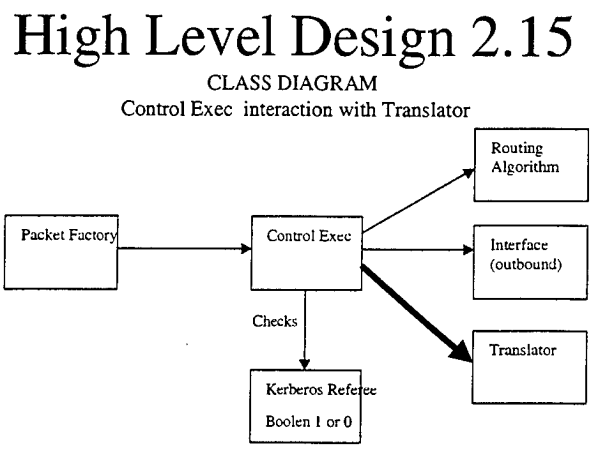


Figure 20. Control Exec Calls the Translator

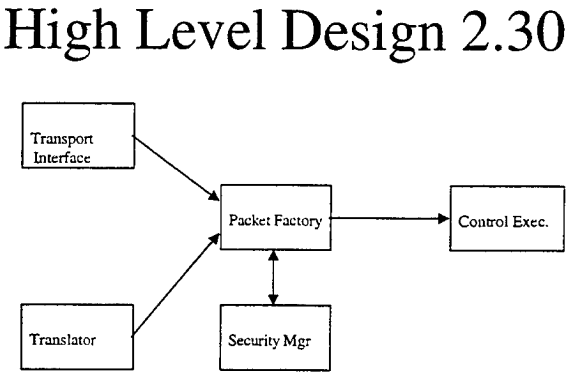


Figure 21. Security Manager Passes Messages with Packet Factory

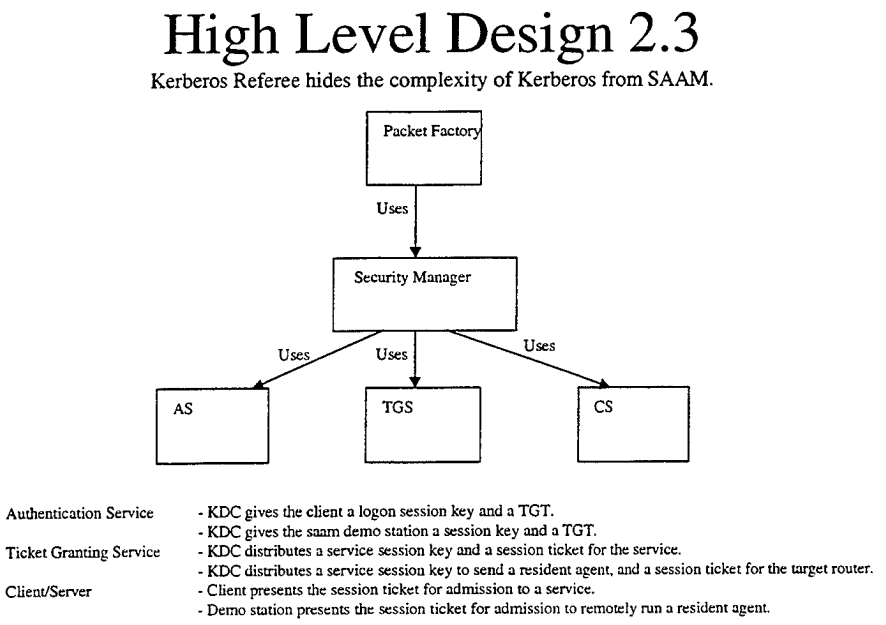


Figure 22. Security Manager Hides Kerberos Details from SAAM

F. PACKET STRUCTURE

The first type of packet is the Encrypted Recognition Key Table.

Figure 23. Packet with Recognition Key Table Encrypted with Trusted Session Key

The second type of packet is the signed signaling traffic

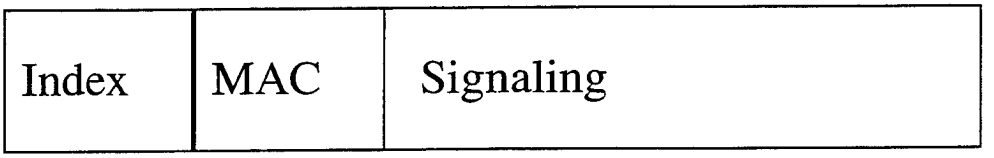


Figure 24. Packet With Signed Signaling Traffic

V. TIME PROTOCOL

SAAM requires time synchronization between nodes for two reasons. The first reason is that Kerberos requires time synchronization for the authentication process to function properly. Without time synchronization, Kerberos is subject to replay attacks. The second and most important reason for time synchronization is that the global recognition key is identified by the current time. Every two minutes, a new key is used. All nodes must have the same clock time in order to use the same recognition key. SAAM requires time resolution to the millisecond, which NTP does provide. GPS provides time resolutions to the microsecond.

A. NETWORK TIME PROTOCOLS

Previous research students at the Naval Postgraduate School (NPS) recently researched the Network Time Protocol. [Hensley and Ludden] Their prototype used NTP, providing great insight into the applicability of ongoing research and development within the SAAM Enterprise system.

1. Description

The Network Time Protocol (NTP) is a distributed computer clock synchronization protocol. NTP is a standard that is implemented by most major Operating Systems, providing a client-server communication hierarchy.

B. NTP SYNCHRONIZATION

NTP uses Universal Time Coordinated (UTC). NTP uses UTC to synchronize "primary" servers via radio, satellite receiver or modem. These primary servers then adjust the clocks of secondary servers/clients. In order to correctly adjust clocks of secondary server over a LAN or WAN, a time offset of the server clock relative to the client clock is computer by the client running NTP. In existence today, there are 79 public primary servers synchronized directly to UTC, located in every continent except Antarctica. There are over 100 public secondary servers synchronized to the primary servers and providing synchronization to more than 100,000 clients and server in the Internet. Additionally, there are an unknown number of private servers utilizing NTP. [Ahmad] Brining up another server is very straightforward.

C. NTP CLOCK SYNCHRONIZATION

The general model for discovering the clock offset starts with a server sending a message that includes its current clock value to the client, which could be another server or workstation. The client records its own current clock value upon arrival of the message. For accuracy, the client has to measure the server-client propagation delay. NTP measures the total roundtrip delay and assumes the propagation times are statistically equal in each direction. [Hensley and Ludden]

Clock errors are due to variations in network delay and latencies in computer hardware and software (jitter), as well as clock oscillator instability (wander). According to NTP documentation, NTP in the majority of cases can keep clock synchronization within a few milliseconds on LANS and a few tens of milliseconds on WANs. [Hensley

and Ludden] This performance may be acceptable for the target integration-testing phase of SAAM.

NTP does not have well established security. The current security in NTP addresses statistical attacks and does not address attacks from a malicious attacker. For example, an attacker can delay time messages so that a target host will get the wrong time, forcing the clock to skew. One idea to prevent this type of malicious attack is to make every host synchronize with a Global Positioning System (GPS) satellite. An antenna would be connected to every computer. This design will prevent the malicious attacker from skewing the clock.

D. NTP IMPLEMENTATION IN SAAM

SAAM Enterprise implementation of NTP would constitute the following overview strategy. A standalone dedicated time server module would be designated as the master timeserver for the SAAM Enterprise. This master server can be a cluster to prevent a single point of failure. The Server NTP time module once incorporated into the SAAM architecture can a be a feature that is enabled by default if it is the only Router within a SAAM Enterprise configuration. This SAAM Server would also be default be its own Kerberos Distribution Center (KDC) if no other SAAM Servers or Routers are available on the Network. Follow on SAAM Servers or Routers then joining the SAAM Enterprise would listen for a broadcast time or unicast time protocol message from the existing KDC/NTP Server. Once the discovery of a master timeserver has been established then the clock synchronization process would immediately take effect as described in paragraph C. NTP CLOCK SYNCHRONIZATION.

E. WINDOWS TIME SYNCHRONIZATION SERVICE

Windows 2000 (Win2K) uses a time service, known as Windows Time Synchronization Service (Win32Time), to ensure that all Win2K computers on your network use a common time. In fact, MIT Kerberos 5, Win2K's default authentication protocol, requires the service. In Win2K, time synchronization is crucial because Kerberos uses workstation time as part of the authentication process. Let's discuss the time service, which complies with the Simple Network Time Protocol (SNTP). (For more information about SNTP, see Request for Comments— RFC—1769.) [Ahmad]

F. IMPLEMENTATION

When a client workstation (i.e., a Windows 2000 Professional—Win2K Pro—machine) boots, it contacts a domain controller for authentication. When the two computers exchange authentication packets, the client adjusts its local time based on the target (i.e., the domain controller's) time. If the target time is ahead of local (i.e., the client's) time by less than 2 minutes, the client immediately adjusts its time to match the target time. If the target time is behind the local time by less than 2 minutes, the client slows its clock over a period of 20 minutes until the two times are in synch. If the local time is off by more than 2 minutes, the client immediately sets its time to match the target time.

Because time synchronization is so critical, the client periodically verifies that its time is in synch with the timeserver. By default, the client performs these checks every 8 hours. It connects to the authenticating domain controller, which is its inbound time partner, and performs the checks using a strategy that seeks to attain a convergence

wherein the two computers are never more than 2 seconds apart. If the local time strays by more than 2 seconds, the client checks its time against the authenticating domain controller more often—in fact, it divides its verifying interval in half, repeating this division until it meets one of the following conditions:

The difference between the local and target is no more than 2 seconds

-The interval reaches its shortest duration (by default, 45 minutes)

-When the two computers' times return to within 2 seconds of each other, the verification interval doubles at each check until reaching the maximum interval of 8 hours. [Ahmad]

The reader may notice that the time resolution of 2 seconds is much rougher than the millisecond resolution provided by NTP.

Note that the SAAM prototype has a scaling factor of about 300:1. For every 300 seconds of prototype operation, a real router could do in about 1 second.

G. TIME SERVICE HIERACHY

Windows Time Synchronization Service uses a hierarchical relationship that focuses on the PDC Emulator at the root of the Active Directory (AD) forest. By default, the first domain controller in a forest acts as the PDC Emulator for the root domain and becomes authoritative for the entire enterprise—an event that the Event Viewer logs in the system log as Event ID 62. You've probably seen the Event Viewer filled with Event ID 62 from the source Win32Time. The description field states, "This Machine is a PDC of the domain at the root of the forest. Configure to sync from External time source using the net command, 'net time /setsntp:<server name>'." In other words, you must configure the PDC Emulator to recognize an external SNTP timeserver as authoritative using the

Net Time command from the command prompt. Type in the following at the command prompt for the syntax.

net time /?

You can use any of the following US Naval Observatory SNTP timeservers:

- tick.usno.navy.mil at 192.4.41.40
- tock.usno.navy.mil at 192.5.41.41
- ntp2.usno.navy.mil at 192.5.41.209

Let's look at the time service hierarchy from the bottom up to see how computers synchronize times and dates with their time partners. Workstations and member servers in a domain use the authenticating domain controller as their inbound time partner. Domain controllers use the PDC Emulator in their own domain as their inbound time partner. The PDC Emulator in each domain uses the PDC Emulator in its parent domain as the inbound time partner, until we reach the top of the hierarchy—the root domain. The PDC Emulator in the root of the forest is the authoritative time server, which you should set manually to synchronize time with an external SNTP time server, as discussed earlier.

One final note: SNTP uses UDP port 123 by default. If you want to synchronize your timeserver with an SNTP server on the Internet, make sure that port is available. [Ahmad]

VI. PROTOTYPES

A. SAAM INTEGRATED PROTOTYPE

This prototype was implemented with Jerome Brock and Joel MacRitchie. The SAAM software code version used was SAAMv1.0May2000. This code contained the integrated modules of previous thesis students, and was optimized by Cary Colwell. This code was noticeably much faster than the code in use last December.

1. Topology

There were four machines used in the topology, a KDC, Bravo, Charlie and a sniffer. We were focusing only on the communication between a Recognized SAAM router with a New Node SAAM Router. A SAAM Server was stood up but not used in our testing. The SAAM server was essentially an extra PC. The standard SAAM demo code of one Server and 2 Routers was used.

2. KDC

The Kerberos Distribution Center (KDC) ran MIT's KDC distribution code. The KDC ran on Linux. The physical PC was down in the laboratory in Span-238. Most of the other testing occurred in the lab upstairs in Span-525. The KDC was stable over several days, serving Kerberos tickets without need for a reboot.

One of the reasons that the KDC was installed downstairs was that we ran out of drive space on our Net1 network upstairs. A new 15GB hard drive was purchased and used in the KDC downstairs. This was plenty of room.

3. JCSI Code

JCSI is Java code that implements Kerberos for clients to request Kerberos sessions from the MIT KDC. JCSI was chosen because it is native Java, just like the SAAM prototype. The JCSI code implements the GSS-API, which is an open standard API, designed for Kerberos. There are two parts to a Kerberized host, the client service, and the server service. Please notice that the Kerberos server service is completely distinct from the KDC.

a. Kerberos Client

The JCSI Kerberos client is the Kerberos principle that initiates the security session. For example, in the New Join Scenario, RouterA would invoke a Kerberos client just after receipt of the JoinRequest message from RouterB.

In our prototype, the JCSI Kerberos client code ran on Charlie.net1.cs.nps.navy.mil. In addition, the client also ran on the PC in the Span-221 classroom during our presentation, for a live demo.

b. Kerberos Server

The JCSI Kerberos server is the Kerberos principle that receives the security session. The server is sometimes called the target principle. For example, RouterB would run the server in the New Join Timing Diagram. In our prototype, the JCSI Kerberos Server was running on Bravo.net1.cs.nps.navy.mil. Ultimately, the client and server were just another object instantiated by the SAAM Translator.

4. Security Manager

The Security Manager was a class in the saam.security package. The Security Manager effectively instantiated the Kerberos client or server, based on input from the Packet Factory. A boolean value passed into the Security Manager constructor is parsed to switch between a server or client. The Packet Factory instantiates the Security Manager.

5. Packet Factory

We found a spot in the Packet Factory to input the constructor for the Security Manager. No more logic was added for our prototype, yet.

6. Packet Sniffer

Once the Kerberos Client was instantiated, the client immediately communicated with the KDC, running the first two Kerberos protocols of Ticket Granting Service (TGS) and Authentication Service (AS). The Client Server (CS) Kerberos protocol began when Charlie (the client) communicated with Bravo (the server). Our topology also included a packet sniffer to capture the packets going between these two.

We hard coded a message to represent the Recognition Key Table, for lack of time to get a real key table. The message read "This is the key table." When encryption was not selected in the code, we were able to see that string in a packet. When encryption was selected, the string was encrypted. The switch is a boolean value passed into the constructor for the Security Manager.

7. Issues

After implementing this prototype, we discovered some major issues that still need to be overcome to fully implement the SAAM authentication protocol.

a. Key Tab File Distribution

The JCSI server requires a key tab file to be on the local c drive. In the prototype, Bravo (the server) already had a key tab file on the c drive. Automatic Kerberos session setup for a new host will now face the difficulty of getting the key tab file in place, if JCSI is used. Windows 2000 may be storing the key tab information in the local registry, but any lookup was invisible to us when we tried Windows 2000 Kerberos.

b. Security Manager Behavior

The Security Manager in our prototype was very basic. The full behavior of the SAAM authentication protocol still needs to be implemented.

B. SUPPORTING PROTOTYPES

The prototypes developed can be categorized into two groups, phase one and phase two. Phase one implemented Kerberos, distributing a Trusted Session key to the Kerberos client and server hosts.

1. Phase One Prototypes

Three different prototypes made Kerberos usable by SAAM. One used Java and a batch file in Windows 2000, the second used C code and called the SSPI in Windows 2000, and the third was an effort to use Java with JCSI on MIT's Kerberos.

a. Windows 2000

Windows 2000 relies on Kerberos realms for authentication. Kerberos is only used with domain logon to Active Directory, that is, only used when there is a logon over the network onto a Windows2000 server. Kerberos is not used for local logon.

We implemented a Windows2000 Domain using Windows 2000 Advanced Server. Advanced server offers some extended features for Microsoft Back office that we did not use. Windows 2000 Server would also work.

When a host logs onto Win2K Server, the Server tries to authenticate with Kerberos. If that fails, then the Server tries to use LAN Manager (LanMan) for authentication. When we installed this OS, we chose to not install NETBIOS because we did not want to allow any hosts to log on without using Kerberos. We chose to not be backward compatible.

(1) Java. A simple Java chat program was adapted from Deitel and Deitel's Book, Java How to Program. In this prototype, buffaloberry was running the chat client and saamwks2 was running the chat server. In the Java code, the line before opening the socket, we forced the KDC to issue a ticket for the target server. How this was done was rather brute force. Java called a batch file, which issued a net use command to map a drive from buffaloberry to saamwks2. The OS took care of issuing

Kerberos credentials when the drive was mapped. The credentials cache on buffaloberry indeed showed that a ticket for saamwks2 had been issued. We examined the credentials cache using a tool called klist.exe from Microsoft. This prototype effectively forced a Kerberos session to be opened, but was not quite the intricate control we had in mind for SAAM.

(2) C code. The SSPI provided function calls to get to Kerberos in Win2K. We were able to adapt the SSPI code that is freely available from the following web site:

http://msdn.microsoft.com/library/default.asp?URL=/library/psdk/secspi/sspiref_00oj.htm

Open the table of contents to the following sample code:

MSDN Library\PlatformSDK\Security\LogonAuthentication\Security Support Provider Interface\Using SSPI\Sample SSPI Code.

After several days of debugging, we finally got this code to work.

The Server code had to be running. The client code could not call the server's OS, but actually needed to have server code. Once the two applications established a socket, Kerberos was used. A ticket was issued and a short session was established and closed. Once again, etherpeek revealed that Kerberos was indeed being called and sent encrypted packets.

A major problem with this prototype was how to pass messages from C to Java. Java Native Interface (JNI) was the method attempted, but we could not get it to work. The complexity of C, followed by the complexity of type marshalling literals through the JNI barrier was a problem that proved to be non-trivial. Given expertise in C, JNI, and Java, this code should certainly work. One drawback to using the

Windows 2000 was lack of platform independence that the current SAAM prototype enjoys with Java.

2. Phase Two Prototypes

Once we could establish a shared Trusted Session Key, the remaining part of the authentication protocol was prototyped. A Recognition Key Table was generated and distributed with the following prototype.

a. Web Recognition Key

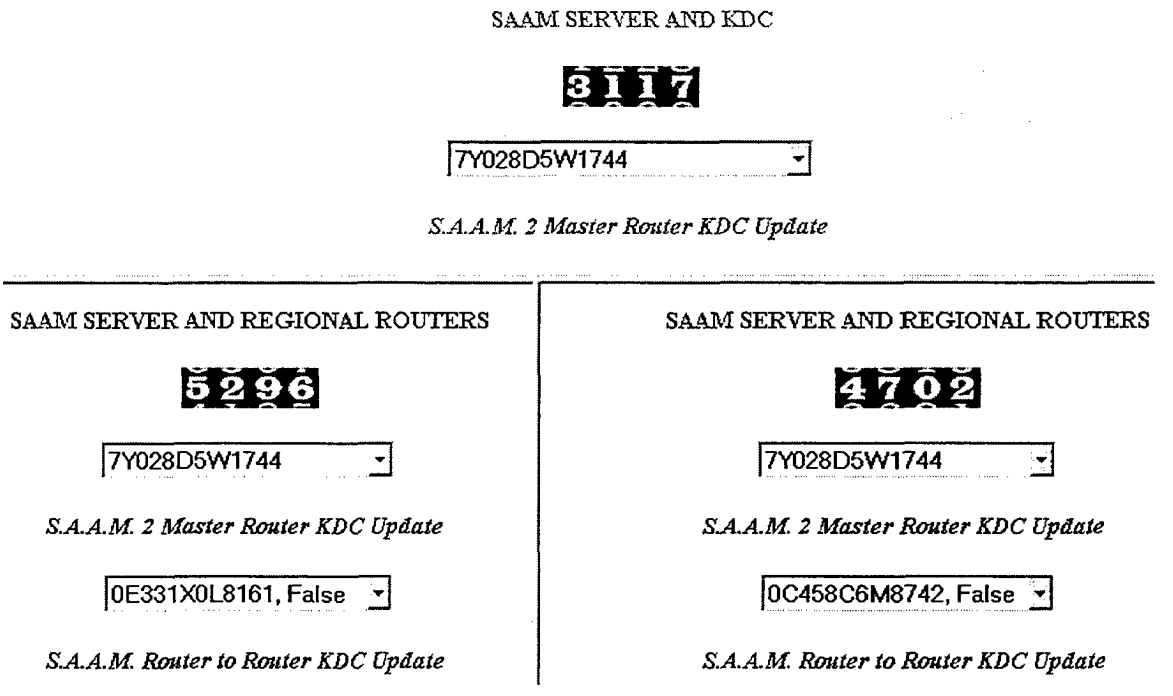


Figure 25. Web Recognition Key

The prototype demonstration for the key distribution table was constructed with the following devices:

a. Windows 2000 Server

b. Internet Information Server (IIS 5.0)

c. Microsoft SQL v 7.0 Server

d. Active Server Page (ASP)

e. Java Scripting

f. Secure Socket Layer (SSL) 128 bit encryption

The prototype web key distribution system demonstrates that functionality of changing Recognition Keys that will occur on a regular basis throughout the twenty-four hour window. The SQL Server generated the Recognition Key Table. A Web server served the appropriate global Recognition key down to each client browser. Four web browsers were opened on one computer screen and a PC other than the SQL server. Each browser was tracked by the web server and served the Recognition key.

Each Recognition Key has a predetermined life of two (2) minutes each. Each Recognition Key is comprised of unique Hexadecimal (32 Digits) representation.

32 Digits = 32 Nibbles = 16 Bytes = 128 Bits

Number of hours /day	24
Number of minutes/hour	60
Shelf-life per Recognition Key	2 minutes
Total number of keys required per 24 hour period:	720 Recognition Keys

Table 2. Design Considerations For the Recognition Key Table

Each key is selected dependent upon the time of the day. The algorithm used for determining the correct key to use for the generation of the Message Authentication Code (MAC) is:

$$\frac{\text{(Current Time – Starting Time)}}{\text{Lifetime of key}}$$

A practical example of the key selection is as follows:

Total Number of keys	720
Starting Time	0730
Current Time	0800

Figure 26. Data for Example of Key Selection

$$\frac{0800 - 0730}{2 \text{ minutes}} = \frac{30 \text{ minutes}}{2 \text{ minutes}} = 15$$

Therefore, the Recognition Key index number 15 out of 720 is to be used in the construction of the MAC. The first fifteen keys are represented in the following demonstration table of Recognition Keys.

Key #	Recognition Key Index	Recognition Key
1	AB71	AB710BCF212187943210323213 45F32D
2	ABE9	ABE984038FCD63981CBD78F898A6D0
3	4038	4038FCD63981CBD78CD63981CBABE0
4	9840	984038FCD78CD63981CBD78CD63981A
5	D639	D63981038FCD68CD63988CD6384038F
6	BCE2	BCE271FA73923BC930AF44BC384038F
7	3840	38403CDE5216AC73BCF8298BCF7208F
8	9040	9040BCDF3840BC84BA28DC82AC2D38F
9	674B	674BC4DF789AC345DCB64BC21DS038F
10	3840	3840BC4590982ADF423BCD32A231138F
11	BC45	BC4537CADD2FF1908403C 384038F3840
12	ABC8	ABC82821090ACBD7898A11FFS0901199
13	D9C7	D9C769828DD579001AA63819AA84B189
14	6AA3	6AA389ABCF43912CDFE80901AF34D117
15	BC80	BC80902734FF268732DAAEB22289871CA

Table 3. Recognition Key Table

The SAAM security manager module will use the calculated key index value of 15 to quickly identify the appropriate key to use with the generation of the MAC. The key and the outbound message are then passed via the MD5 algorithm to produce an MAC. The corresponding receiving SAAM Security Manager would quickly look at the time of the inbound message traffic to determine the appropriate key to use from its own copy of the SAAM recognition key table. Once the key selection has been completed at

the receiving-end SAAM security manager, a quick security validation process is conducted to determine if the inbound message is valid or to be discarded. The incoming message includes the key index that was used in the construct of the outbound message traffic, if the 4 digit key index attached to the message does not match the calculated key index derived by the in-bound security manager or the previous key index or the next key in the table then the message is discarded. If the message key index is valid then the message is extracted and run via the MD5 algorithm along with the calculated key. If the derived MAC matches exactly to the MAC of the inbound message server then the message is routed correctly via the SAAM OS.

3. Full Protocol Prototypes

A full prototype of the security protocol was developed rapidly in Visual Basic (VB). However, integration with the run-time SAAM Java prototype was not completed. Writing the Security Manager in VB was very fast for the author, compared to the difficulties encountered with Java and C above.

a. Visual Basic

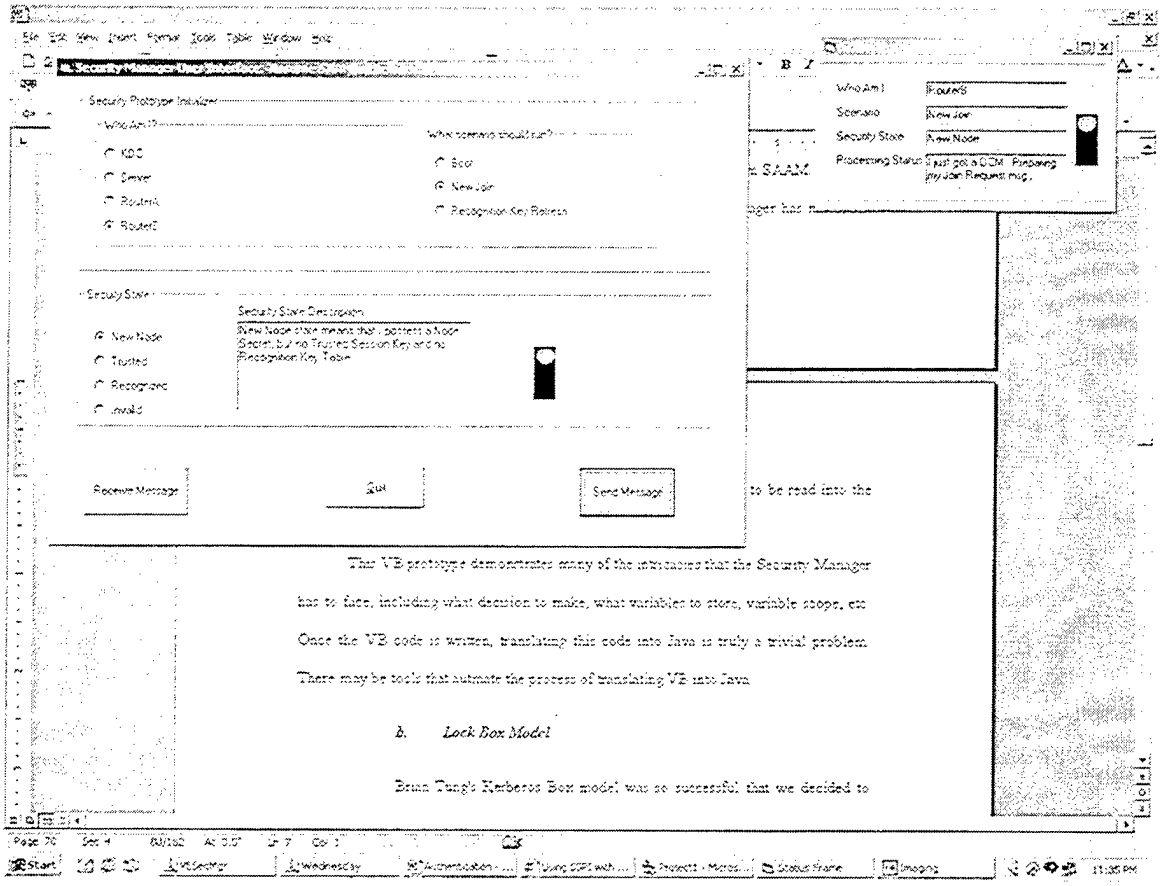


Figure 27. Security Manager User Interface in VB

A major assumption is made for this prototype. It is assumed that SAAM is the user, literally pushing command buttons and filling in answers to questions when prompted.

Security Manager Prototype Flow Diagram

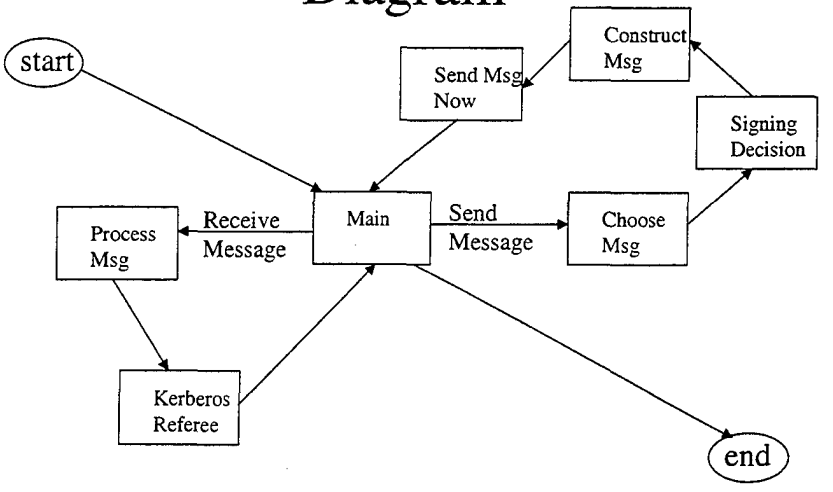


Figure 28. VB Forms Diagram

The flow diagram above shows how the VB forms relate to each other. The Main form is the first GUI that SAAM encounters. The initial state variables are assigned to the Security Manager based on inputs from SAAM. Three questions must be answered. (1) Who am I? (2) What is my current Security State? (3) Of the three scenarios for this security prototype, which scenario is currently running? These three answers must be stored. There is a tray, which holds these three state variables, displayed in the form of a background form in the upper right hand corner of the screen.

SAAM then tells the security manager to do one of two things next. Either send a message, or receive a message. The Security Manager then processes the remaining steps, automating as much as possible with minimal participation from SAAM. Most of these other forms are information of what decisions the Security Manager has

made, and what is going to happen next. Finally a result is returned to SAAM, to be read into the Packet factory for further processing.

This VB prototype demonstrates many of the intricacies that the Security Manager has to face, including what decision to make, what variables to store, variable scope, etc. Once the VB code is written, translating this code into Java is truly a trivial problem. There may be tools that automate the process of translating VB into Java.

b. Lock Box Model

Brian Tung's Kerberos Box model was so successful that we decided to implement the entire protocol using lock boxes. This was the old fashioned way to demonstrate a new protocol, with people standing around a table, using boxes and paper messages to trace out the protocol. No computers were used, but concepts were solidified through these exercises.

THIS PAGE INTENTIONALLY LEFT BLANK

VII. FUTURE WORK

Concerning the JCSI prototype, the following work remains to be done.

1. Tighter Integration with Packet Factory.

The Packet factory is highly threaded. The exact location of where to add calls to the Security Manager still needs to be done. Prof. Xie gave suggestions about where to make calls within the Packet Factory. A detailed understanding of the code inside the packet factory is needed. More time is needed to add improved logic between Packet Factory and Security Manager.

The behavior of the Security Manager is already developed in VB. Translating this VB code directly into Java would help. Each VB form can become a Java Class file. VB is taught in the ITM curriculum. A person with a background in VB could translate the VB Security Manager into Java, as an exercise to learn Java. Translating the Security Manager into Java could be a Java project in one of the Java classes taught at NPS.

2. Implement in/out of Band Model.

Our implementation was entirely out of band of SAAM for several reasons. The first was that JSCI was tightly integrated with IPv4. The second reason was that there was no flow id assigned for security protocol messages, and the current SAAM prototype only routes by flow. Further work on what parts of security are in-band should be done. For example, should the KeyTableResponse message be an in band message?

3. Recognition Key Table.

With more time, a full recognition key table can be created and sent over the Client Server session. Two efforts were ongoing in parallel to make this happen. One was the SQL generated key table, and the other effort was a native Java key table. Both

efforts completed a key table of 720 keys that were 128 bits in length, but the key table was never fully integrated into the protocol.

The following general ideas remain as future work.

1. RIDLR is an acronym, which stands for Reconfigurable Intrusion Detection System. RIDLR is an ongoing project with the Information Warfare students working with LT Buettner, military IW instructor. RIDLR should be made into a resident agent on SAAM, for proof of concept. A future thesis student could be in the world of SAAM and the IW lab.

2. Security has been incorporated into NTP. A review of the articles on Dave Mills' web site, http://www.eecis.udel.edu/~mills/reports.htm, would help to evaluate the secure time synchronization we use in this protocol. The native NTP security could also be incorporated into the SAAM security protocol.

3. How much time is saved by the quick reject behavior? The quick reject behavior occurs when the receiver of a signed message quickly evaluates the key index to see if this is an expected key index. If not, then the entire message is rejected before the hash is done. Once this is fully built into a prototype, sample data can be collected to see how much time is saved. The real question to ask is how many CPU cycles are saved by quick reject? It might be more difficult but not impossible to measure the CPU cycles saved.

4. Kerberos is not the only possible answer for establishing a trust. IPsec could be used instead of Kerberos.

APPENDIX A. ORGANIZATIONAL BEHAVIOR EFFECTS OF SAAM

A. CHAPTER SUMMARY

This chapter is a case study on the effects that SAAM can have on organizational behavior. A Navy command is analyzed before SAAM is applied. The same organization is analyzed again after SAAM is applied. Two environmental factors that influence this study are the Military Technical Revolution (MTR) and the 1996 Telecommunications Act. While many papers have been written from the strategic vision level for the entire Navy organization, this case study is unique because it addresses a small tactical level organization. The key learning points listed at the end of this chapter summarize the important aspects of this case study. Any active network would probably produce the same results. SAAM is used here as an example of an active network.

B. STATEMENT OF PURPOSE

The purpose of this case study is to determine the organizational changes that are likely to take place due to SAAM. The following case study describes a fictional Navy command, called the Center, which uses computer-networking technology that is common for the year 2000. An expert system called OrgCon is used to help analyze the organization in both the before and after scenarios. The benefits and limitations of the IP architecture will be explored. A new network protocol will be integrated into this organization, to allow quality of service (QoS) guarantees over the network. The resulting network and organization changes will be defined.

C. CHAPTER DEFINITIONS

The term equivocality is used often in this chapter. High equivocality in the environment means that you don't know what you don't know. In the case of cyber warfare, capturing the techniques of a hacker can be a situation of very high equivocality. This is like finding a needle in a haystack. Technology changes so rapidly that the term "web time" is used to describe the compression of pre-Internet business with current electronic commerce. This time compression ratio is about 7:1. For every 7 years of old style business production, e-commerce business produces the same amount in 1 year. Defending against a hacker is like finding a needle in a haystack, and the haystack is constantly being shuffled.

D. CASE STUDY ASSUMPTIONS

The following assumptions have been made to frame this case study.

The lack of promotion opportunities, command opportunities, and competitive pay are some of the disincentives that the Navy has as an organization. The rewards system for Information Technology (IT) careers in the Navy is no longer dysfunctional in this case study. [Kerr] In this case study, assume that the best and the brightest IT people are rewarded, producing incentives to seek to enter and stay in the Navy.

The question of specialist versus generalist has been solved before this case study begins. The fine balance has been reached between daily Naval operations, and the infusion of relevant IT into every part of the Navy organization. IT has become like reading, writing, and arithmetic. Everyone in the Center is IT literate. That is not to say

that everyone is a superstar, but everyone is familiar enough with network technology that they can quickly learn new technology that is related.

In this study, the number of personnel is held constant through before and after analyses. Navy commands like the Center in this study can realistically expect an increase in the number of personnel because the Navy is just beginning to implement an Information Operations (IO) strategy, guided by the vision of network centric warfare. [JV2010] No build up will take place in this study, in attempt to focus in on the organizational changes that occur due to SAAM.

All major Internet Service Providers (ISP's) support SAAM worldwide. SAAM is enabled just as any routing protocol is enabled in routers today. Assume that SAAM has been bought out by a large company that manufactures routers, and SAAM is now available as a feature throughout the Internet.

Assume that SAAM is already a secure system for this case study.

E. THE CENTER BEFORE SAAM

The Center invested in a Local Area Network (LAN) upgrade only two years ago. LAN bandwidth locally was increased from 10Mbps to 100Mbps, a full order of magnitude change. However, the latest applications were already stressing the throughput on this network, due in part to the party line nature of Ethernet. As more and more people join in on the party, there is more talking at once, known as collisions.

In the Center, the increase in network collisions was not due solely to an increase in number of computers on the LAN. In this case, the newest applications are demanding more from the network. Multicast streaming video, video conference calls, remote

network monitoring agents, and voice over IP are just some of the applications that have bogged down the LAN in recent months. The topology of the LAN Before any changes are made is shown below.

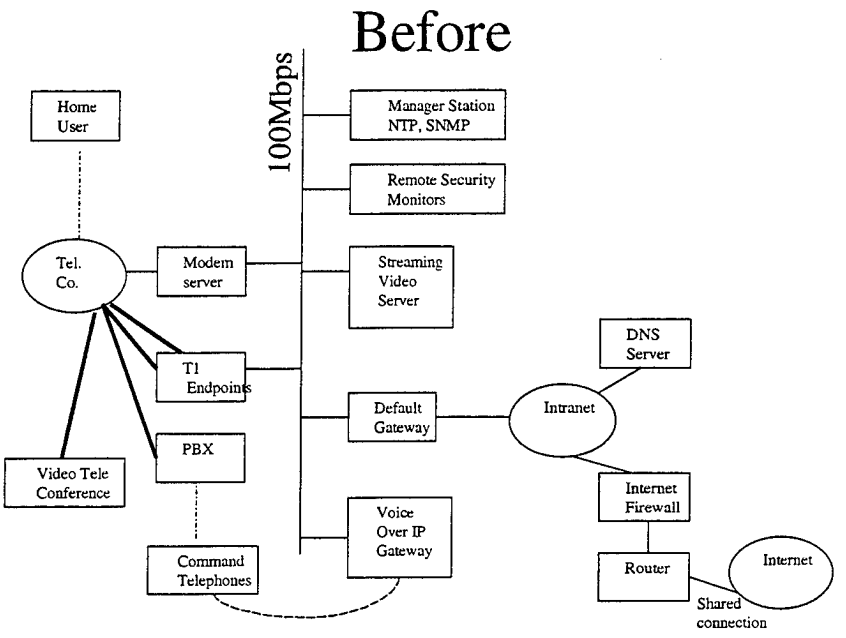


Figure 29. LAN Before SAAM

Meet Luke Ellington. He's the manager of this department, which is at the heart of the business operations. That is why this department is called The Center. As a Navy command, all the workers in The Center are well acclimated to the traditional machine bureaucracy organization found everywhere in the military. The entire department consists of only ten people, organized into two divisions. Bob is the division officer of the computer operators. All people in this division are active duty, bringing with them a range of current skills in the computer networking industry. Alice's division is a mix between active duty, civilian, and contractors. Contractors are hired for projects where

existing expertise is lacking. Contractors also train others during development and installation of a new system. The organization chart of The Center is as follows.

Before

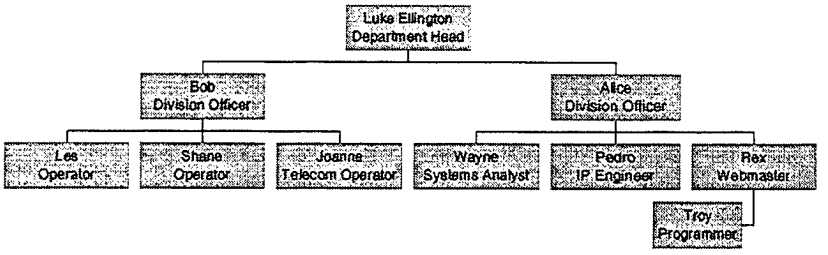


Figure 30. Org Chart of The Center – Before Scenario

The main business process of The Center is to provide network security support to Naval commands, including almost every command in the Navy. The number of computer user seats in the Navy is in the range of 400,000. The potential for security breaches in this large enterprise is the motivation for standing up The Center, which is only five years old.

Most Navy commands have a deep history and experience to pull from in times of crisis. Information technology has been dominated by the private sector for less than two decades, a short time compared to the US Navy's 200+ year history. Tradition runs strong in the Navy, but this giant organization saw the need for a forward thinking networking command to help deter the threat from the enemy over Internet. This new

form of national defense is called information operations (IO), which is a more strategic view of all things in the information warfare (IW) arena. And so The Center was born, by order of the Chief of Naval Operations (CNO). Though young in years, The Center has never ceased to be in the limelight.

There is an organizational tendency for LAN administrators to become "Data Czar's," hoarding massive organizational control. At times, the LAN administrator has control over files that used to be readily accessible in a physical file locker. If the LAN administrator is holding too many keys, then the organization suffers when individuals cannot get their data because the LAN administrator is busy fighting fires elsewhere. Over-ambitious LAN administrators exhibit this behavior when the organization rewards it. This breeds a lack of trust and rice-bowl mentality, where knowledge really is power. The center is a collection of well-seasoned IT professionals, chosen in part for their personal grace under pressure and teamwork. The center has been able to avoid this Data Czar mentality, through professionlization. Professionalization means that employees have completed a rigorous standard education of many years, such as medical doctors. People at the center have seen this before, share a common business history in a sense.

1. Environmental Factors

The Center is a unique command in the Navy. The very existence of the Center is due to the Military Technical Revolution (MTR). While most admirals today have no direct experience at using the Internet as weapon, most do realize the importance of a solid Information Operations (IO) infrastructure in the Navy. The fact is that the civilian population in the USA was exposed to direct attack from any cyber-terrorist, and the

military was not capable of defending our population at all. The figure below demonstrates this idea.

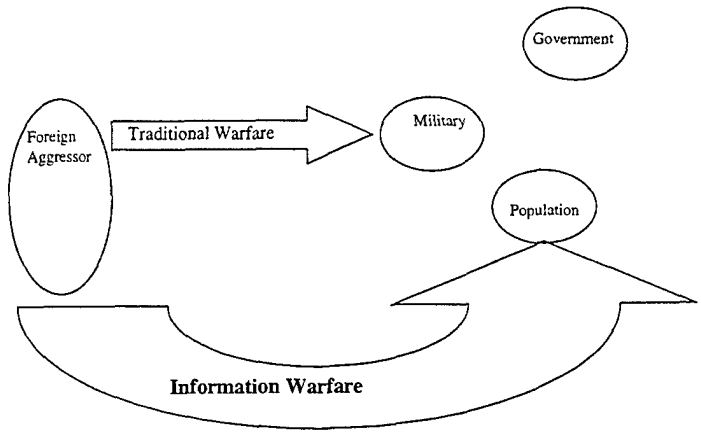


Figure 31. Military Without IO

The strategic view is to maintain Information Superiority, forcing the aggressor to go through the military before attacking the population.

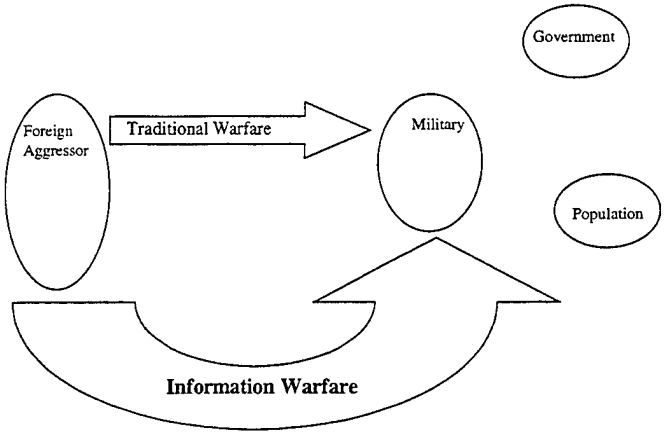


Figure 32. Military With IO

The Center is one of the commands created to make this strategic view a tactical reality. That is why the Center has an increasing budget. When most commands have ever shrinking budgets, the Center is actually expanding. Still, the budget at the Center undergoes careful scrutiny to ensure that spending is done wisely. This budget increase is nothing compared to the order of magnitude difference seen during the military build-up during the 1980's.

The spotlight is another environmental factor. Commands like the Center are in the spotlight from above, garnering high visibility from the highest-ranking officers in the Navy. The theory of social facilitation applies to center, even from an organizational level rather than individual. When performing a well-known task and the Center is aware that it is being watched, the task is done extremely well. When performing a task that is not well know and the Center is aware that it is being watched, the task is not done as

well as it would be without an audience. Learning new technologies on an operational network is difficult to do.

The final environmental factor of great influence to SAAM is the 1996 Telecommunications act. This legislation basically makes it legal for the telephone, cable and Internet companies to compete in any of these three industries. [Lewis] Thus, free competition legally exists between voice, video, and data. The legislation applies most directly to private companies. Since the military now uses commercial solutions, COTS, the Telecom Act affects the Center. Specifically, the Center is currently paying bills for telephone access, dedicated data lines at T-1 speeds, and a separate bill to connect to the Internet. It would be ideal for any customer of telecommunications services to combine all these services to pay a much smaller bill.

2. Analysis of Before Scenario

The following is taken from the results OrgCon, the expert system.

The quality of people working at The Center is very high. Postgraduate education levels are greater than 60%, including all the enlisted personnel. It is common to see senior chiefs (E-8) with master's degrees in technical and managerial fields. The level of trust within The Center is very high. The group is small and highly educated, giving an influence towards a professional organization. Small mistakes are tolerated, so long as the mission is not impacted. But as any Navy command, there is not much room for failure. Luke's leadership style is basically risk avoidance. Still, this is not quite as extreme as the zero defect mentality that pervades other Navy communities.

The environment in which the Center operates is to face the constant security threat from the global Internet. State sponsored hackers and phreakers are constantly innovating new ways to snoop on Navy commands. The Center must be agile enough to out step this threat, and vigilant enough to manage any attacks that could not be prevented. There are not many written Standard Operating Procedures (SOP's), although the few pass down items of mundane computer operations are written and followed. The people in The Center are given enough room to develop new and innovative solutions to cutting edge technology problems.

The developmental climate is characterized as a dynamic, entrepreneurial and creative place to work. The tendency for people to stick their necks out and take risks is held in check by a risk averse Navy culture. The glue that holds organizations together is commitment to experimentation and innovation. The emphasis is on being on the leading edge. Readiness for change and meeting new challenges are important. The Center's long-term emphasis is on growth and acquiring new resources. Success means having unique and new products or services and being a product or service leader is important. The Center encourages individual initiative and freedom.

The Center has a high level of trust and a group climate. High morale is one element of group climate. Highly equitable rewards within The Center drives the climate towards a group climate. The Center has a low level of scapegoating, which is also conducive to the group climate.

Since Luke and his boss are constantly under the scrutiny of Washington DC, he has a preference for being very involved in gathering and using detailed information when making decisions. A high preference for micro involvement characterization is

appropriate. Management is risk averse, which is one of the characteristics of a manager with a high preference for micro involvement. Because The Center is a small organization the preference for micro involvement will be higher than it would otherwise be.

The Center's strategy can be characterized as an analyzer without innovation strategy, which is an organization whose goal is to move into new services only after their viability has been shown, yet maintains an emphasis on its ongoing services. It has limited innovation related to the production process; generally an analyzer without innovation does not have product innovation. Most new services that The Center provides come from an external order from a higher authority, rather than from demand from the customers or from innovation in house.

The capital requirement of a hacker is not high. However, the capital requirement for the Center is high because it must purchase enterprise level security systems, such as complex intrusion detection system. This high capital requirement is consistent with an analyzer without innovation strategy. Any hacker who wishes to infiltrate the US Navy can do so with a home PC and a connection to the Internet. This is now in the price range of below $500. With a very routine technology, new products for new customers are not very likely, although the firm can copy a few products. Therefore, strategy is likely to be analyzer without innovation. With a concern for high quality an analyzer without innovation strategy is a likely strategy for The Center.

The Center has both a routine technology and a highly equivocal environment. A more non-routine technology is a better fit with an equivocal environment. A routine technology produces services efficiently which are standard and without variation. In a

highly equivocal environment such as combating hackers on the Internet, it is likely that the Navy commands who are customers will demand variation in the product and service characteristics. Hackers are likely to introduce new products, modify footprints, etc. Further, in the equivocal environment, large changes can come from unforeseen actions by hackers, foreign governments, and breakthrough innovations from our own Silicon Valley available to the world at large. A more non-routine technology will be required to adapt to the unknowns and changes of an equivocal environment.

The Center has both a routine technology and a high requirement for product innovation. This situation must be changed; a routine technology will not support high product innovation. A routine technology yields standard services with low variation. The need for service innovation creates a mismatch. Service innovation will be difficult to manage, expensive and inefficient. For service innovation, a more non-routine and adaptable technology is required. Of course, the Center does not have the option of changing to a different environment where less innovation is required and a routine technology is suitable.

The Center has a developmental climate. This is a mismatch with the leadership being risk averse. Leadership in this sense includes Luke and his entire chain of command. A development climate is relatively flexible and externally oriented, characterized by low conflict, low resistance to change, and high leader credibility, among others. Risk averse leaders generally avoid uncertainty and generally also minimize change. The risk averse leader will not be comfortable in a developmental climate and may introduce more control to reduce the uncertainty.

The Center tends to move toward a developmental climate. This is a mismatch with a highly routine technology. A developmental climate is flexible and has an external orientation. It has low resistance to change, low conflict and high leader credibility, among others. A routine technology is more compatible with a climate of stability and there is not much change. The focus is on running the routine technology. A developmental climate can support a more non-routine technology where adaptation and variation are the norm.

Since the set of variables in the environment that will be important is not known and since it is not possible to predict what will happen, no efficient rules and procedures can be developed, which implies that the Center's formalization should be low. A developmental climate in the organization requires a low level of formalization.

When the environment of the Center has high equivocality, high uncertainty, and high complexity, coordination and control should be obtained through integrators and group meetings. The richness of the technical media should be high with a large amount of information. For example, large screens and visualization should be present in the meeting room where executive decisions are made. Incentives must be results based. When the organization has a developmental climate, coordination should be obtained using planning, integrators and meetings. Incentives could be results based with an individual orientation. An organization with a developmental climate will likely have to process a large amount of information and will need information media with high richness.

Top management, Luke, should make many decisions. However, many individuals should be involved in gathering information and implementing those

decisions. Top management should gather information, make decisions, and manage implementation. Top management should give direct orders to achieve the required coordination among the operations and activities.

The following organization misfits are present in the Center before SAAM is applied:

Current and prescribed configurations do not match.

Current and prescribed complexities do not match.

F. MOTIVATION FOR QUALITY OF SERVICE OVER IP

Sidgmore's Law – the number of packets on the Internet backbone doubles every three months. This means that the number of packets on the Internet backbone increases by about 16 times per year. [Lewis]

Most of the intrusion detection systems run over the shared line, which goes out into the Internet. The LAN diagram above shows this shared line connecting the router to the Internet. Some critical applications currently run over leased lines. The leased lines go out to the local telephone company at a recurring cost of $1000/month. [Redshift] The Center has four leased lines, one of them being the T-1 connection out to the Internet. The recurring cost of the other three lines has come into question by Luke's superiors. He has financial pressure to relinquish one or two of those leased lines.

At the same time, Luke's technical workers tell him about application failures due to network congestion. Applications that monitor other networks need to have a clean connection to the customer sites, and that connection must be in band with all the other IP traffic to be effective. The monitoring software must flow along with the regular traffic.

In order to find the bad fish, you must set your decoys to swim with the whole school. Routers are dropping LAN packets during times of heavy congestion.

Luke is offered several options:

- Eliminate two leased lines and forget about technical requirements for more bandwidth.
- Buy more leased lines and forget about senior input.
- Witch-hunt. Find the offending applications on the LAN and eliminate all applications that are not specifically defined as mission critical. Luke has recently heard that the streaming video his crew has constructed on how to configure a firewall has won praise from the Navy commands the Center services, and this is reflected in comments from Admirals above him. Yet Luke also knows that this is one of the applications that is eating away at his precious bandwidth.
- Institute a Quality of Service system over IP.

Luke chooses plan d, which is the best balance of budget pressure from above with bandwidth demands from below. Choosing a QoS over IP solution makes sense for the center. From the financial perspective, the Center will eliminate the recurring monthly cost of $3000 per month. There will be a one-time cost to install the QoS system, but it will take about six months before the breakeven point is crossed and the new SAAM system pays for itself in cost savings.

Technically speaking, the level of service on the old dedicated lines is very high. The videoconference node on one end receives real time traffic from the other end. There are few blips and static. Once in a while, the connection is dropped for one reason or another. When that happens, the screen freezes. But the sound and video are pretty clear.

This is contrasted with the Voice over IP (VoIP) system, which has a noticeable decrease in service level. There are occasional blips of silence mid conversation. Sometimes, during mid day when everybody is busy at work, the VoIP system is almost unusable because of congestion. So the question that Luke must answer is how to migrate his video conferencing application to the IP world, while still maintaining the high quality of service perceived by the end users.

The current Internet uses one service level model for all IP packets. Best effort service means that all packets on the Internet have equal priority. When there is an overload of packets, the packets are dropped. The suggestion to use Asynchronous Transfer Mode (ATM) was rejected based on price and the lack of ability to 'take care of the last mile' and offer ATM QoS all the way into the home. Luke often finds himself doing work on nights and weekends from home, sometimes speaking with a force commander in a time zone 12 hours away. There are four QoS over IP models that Luke has available to him: (1) Integrated Service, (2) Differentiated Service, (3) Multi Protocol Layer Switching (MPLS), and (4) Server and Agent based Active Management (SAAM).

G. THE DECISION TO USE SAAM

Integrated Service offers a guaranteed class of service. The problem is that many telecom carriers do not offer Integrated Service because there is no way to meter and appropriately charge who is using the bandwidth. On the other hand, differential service looks promising at first glance. Luke will be able to set up his applications to be either premium service, assured service, or best effort service and pay an appropriate price tag for each. MPLS does much the same thing, with per-class guarantees and routers that

dynamically change their own state. The price quote below is an example of how the local Internet Service Provider (ISP) and/or telephone company could price the different service levels in differentiated service.

SERVICE	PREMIUM SERVICE	ASSURED SERVICE	BEST EFFORT SERVICE
BANDWIDTH REQUIREMENT	128 KBPS FRACTIONAL T1	384 KBPS FRACTIONAL OF T1	768 KBPS FRACTIONAL T1 (REMAINING BANDWIDTH)
ISP CHARGE	$3599.95 MONTHLY $1495.95 SETUP	$1999.95 MONTHLY $1495.95 SETUP	$659.95 MONTHLY $1495.95 SETUP

Table 4. Pricing Model for Differentiated Service

At first glance, this looks promising to Luke. However, the problem is that differential service only makes per class guarantees, not per connection guarantees. When he runs the video teleconference (VTC), he does not want that to lose out to other high priority traffic, such as the remote intrusion detection systems. The other technical problem is that when all of the premium service bandwidth is in use, the network begins to drop premium packets and quality is deteriorated.

Luke looks to SAAM for a solution, and finds that he can have per connection guarantees. Each application can enjoy high QoS as needed, with higher priority applications always gaining the bandwidth they need. This means that he can have his network monitoring software running in the background, while holding a video teleconference at the same time. If somebody begins to place a VoIP phone call, they will get the remaining bandwidth available, and the phone call will not impact the video teleconference at all. SAAM guarantees an end-to-end connection for each application. The pricing model for SAAM is shown below.

APPLICATION	VTC	VOIP	NETMON SOFTWARE
BANDWIDTH REQUIREMENT	128 KBPS	384 KBPS	768 KBPS (REMAINING BANDWIDTH)
ISP CHARGE	$2599.95 MONTHLY $1495.95 SETUP	$999.95 MONTHLY $1495.95 SETUP	$659.95 MONTHLY $1495.95 SETUP

Table 5. Pricing Model for SAAM

The price to institute SAAM is lower than any other option and each specific application can be guaranteed the required level of service. Because SAAM runs over IP, there is no wasted bandwidth. As each application 'hangs up' the bandwidth is re-allocated and not wasted. Recall that all major phone companies and ISP's support SAAM today. Luke will be able to run video teleconference (VTC) software and get a good connection all the way to his home PC now.

The figure below summarizes a comparison of the QoS over IP options available.

	WIDELY SUPPORTED	PER-CLASS GUARANTEES	PER-FLOW GUARANTEES	ROUTER OVERHEAD LOCATION	FAIR PRICE FOR BOTH TELCO AND CONSUMER
INTEGRATED SERVICE	NO	YES	NO	INSIDE NETWORK	NO
DIFFERENTIATED SERVICE	YES	YES	NO	INSIDE NETWORK	YES
MPLS	YES	YES	NO	INSIDE NETWORK	YES
SAAM	YES	NO	YES	OFF NETWORK INSIDE SERVER	YES

Table 6. Summary of QoS Options

Luke decides to implement SAAM in his network.

H. MIGRATING TO SAAM

Managing the issues involved in the specific migration should begin with the following generic checklist. This checklist is only a starting point and should be tailored to meet the needs of the specific organization.

Checklist for Migrating to SAAM

- Identify applications that require service level agreements (SLA's). Some of these applications might include voice over IP, fax over IP, video conferencing, streaming audio and video, virtual private networks, hosted applications, e-mail, messaging applications, content, ftp with verified sent, and network games. [Xacct]
- Assign importance to those applications.
- Training. This can run in parallel with other tasks.
- Shop for best value on SAAM. Which ISP offers the best product for this organization at the lowest cost? Set up formal service level agreements with the ISP or local telephone company.
- Conduct a cost benefit analysis to determine the break-even point.
- Implementation.
    - Perform upgrade in house or outsource? A good professional IT staff will be capable of the upgrade in house. The Center would upgrade in house.
    - Test roll out in a test lab first. Red teams the test lab. Try to imagine what the very worst nightmare of an enemy. The red team should make us sweat and work hard.
    - After action review.
    - Roll out SAAM for real. Start with non-operational subnets, such as the administrative LAN. Next, enable SAAM on the operations subnet, the backbone, the choke router and finally out to the first hop on the Internet.

- Analyze progress. Keep the feedback loop running throughout the implementation.
- Allow time for SAAM to stabilize. There will be some growing pains in the organization as people become used to SAAM, especially the first few hours, as the routers in the Internet adjust.

I. THE CENTER AFTER SAAM

The ability to guarantee QoS affects the local network in the Center. The applications that need bandwidth now get it when and where they need it. The Streaming video server begins to be used far more than before, as more and more Navy personnel discover this service available to them. Training on firewall setup is crucial throughout the Navy enterprise. This streaming video server has been changed to a beefed up platform to service all the hits. The video teleconferencing is going well. With no degradation in service, Luke has managed to migrate the video teleconference applications and save that recurring cost of the dedicated lines. The Voice over IP service has been rolled out throughout the center now that the quality is about the same as a regular phone call. In the past year, only about two VoIP phone calls have been dropped, which is comparable to the statistics of using the old phone company lines. Once again, the cost savings is enormous. Even though the Center had autovon available to them, the quality was so poor for phone calls to long distance sites throughout the world. The new VoIP solution was saving the Center some $2000 a day, every 24-hour period. Also, the main service provided by the center no longer blinked out during times of high Internet

congestion. The monitoring agents maintained an awake state straight through the year, with 99.999% uptime.

Perhaps the most notable change in the network is the creation of a practice lab. This reproduces the operational network. The practice lab is for development and integration of new tools. For example, SAAM was first integrated on the practice lab. The practice lab is expected to fail occasionally, so no operational systems rely on it. In fact, the practice lab is essentially an isolated network, allowing for traffic to the Internet. The practice lab was also deemed necessary because of the high equivocality in the environment. As new IO attacks are discovered, the details of defending against them are uncovered in the practice lab. The figure below shows changes made in red.

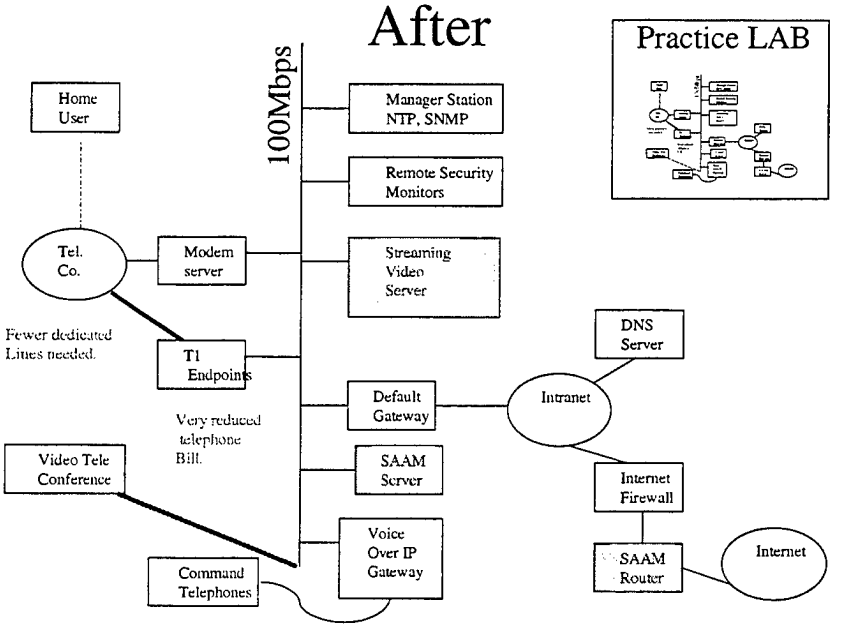


Figure 33. LAN After SAAM

In addition to the technical changes that have taken place, the organization has now changed. The overall configuration has moved to a simple configuration. All people

in the Center now report directly to Luke. This flattens out the organization, allowing for fast reaction to new problems that arise in the environment. This increases the direct span of control for Luke, but it is only ten people, which is still a very small organization. Also, the people in the center are highly trained professionals in this particular field. Strict rules and regulations can hinder their progress more than help, because the working core best understands the solutions to problems.

Two billets that are interesting to note are the IP Engineer billet and the Telecom billet now overlap so much that they could be considered one job. This is demonstrated in the Org Chart below, showing Pedro right next to Joanne. Both billets now run the combined Telecom and IP system. This is a potential source of friction because an organizational boundary has been removed by SAAM. However, no jobs need to be eliminated. Filling the Telecom/IP Engineer billet will need redundant personnel, because Telecom and IP have become such a critical function to the Center.

After

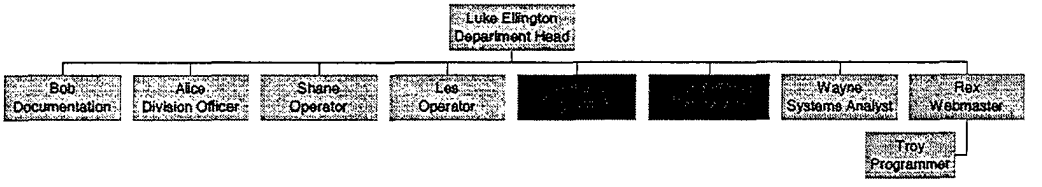


Figure 34. Org Chart of the Center from External Point of View

This organizational chart is still used when reporting to senior officers or commands. However, the internal organization in the Center has really moved more toward a simple organization. The reason for this change is due primarily to the high equivocality in the environment. This Navy command has become like an Internet company, able to change direction to a new product, and able to move forward in that direction at a high velocity. The environment demanded such a rapid pace, and the bureaucratic structure did not easily accept rapid changes.

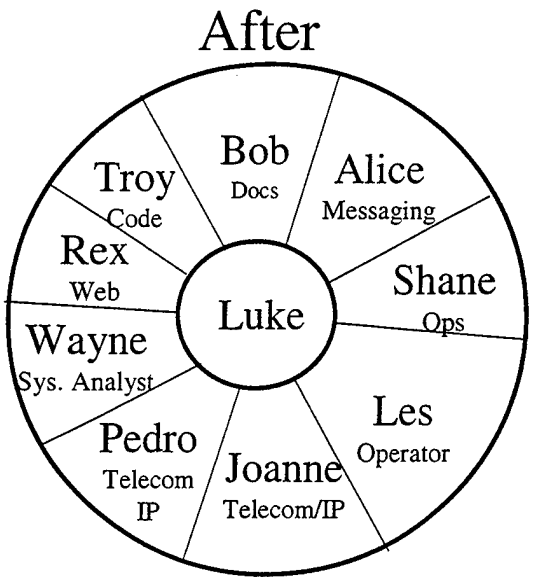


Figure 35. Day-to-Day Working Organization After SAAM

After the advanced network is applied, the organization functions more like a team. Each person brings with them a technical specialty to support Luke, the decision maker. Problems are solved with ad hoc solutions, rather then pre-programmed answers. Agility is gained from this organization. Also, this organization is only as strong as its weakest link, so there are strong group norms to work hard and keep Luke on top of the latest and greatest technology.

This is a cultural change from the external Navy environment, but different community cultures in the Navy do exist and thrive. For example, the culture within an air wing is almost a professional bureaucracy, with each highly trained pilot trusted to make the proper decisions while flying in combat. Without this dynamic culture, the Center could not survive. This disparity between the Information Warrior culture and the Navy Line culture will continue to be a source of friction. Hackers are not bound by

bureaucracy. On the other hand, hackers do not have the benefits of a large standing bureaucracy to draw resources, knowledge, and experience from. The Center is in a unique situation.

J. ANALYSIS OF THE CENTER AFTER SAAM IS APPLIED

The following is a discussion of results from OrgCon.

The personnel working in the center are highly professionalized. This still has not changed, and all the related factors remain the same. The level of trust remains high and scapegoating remains low.

The issue of risk taking has been addressed. The Navy enterprise is a risk avoidant culture, but the Center needs to experiment and innovate in order to sufficiently defend other Navy commands. This paradox was the thinking behind the practice laboratory. The practice laboratory offloads risk. Now the personnel can try new products and services for evaluation quickly, without incurring additional risk to operations. The practice laboratory can also be used for self-training on specific technical issues. The practice laboratory also facilitates the current operations routine technology of the center with the uncertain environment. This situation can cause problems for which a more non-routine technology is better.

The old written SOP's were essentially useless. Pass down information is more reliable because of the short time in which a pass down can be written. The information is still current. This is in contrast to the out of date information in SOP's, which took a longer time to develop. The external Navy environment will want to see SOP's, enforcing this expectation with IG inspections. The Center now has a large manual that is

maintained and kept up to date. It is called the SOP manual, but it is not a traditional Navy SOP. Rather, it is a modular collection of pass down items, updated almost daily. As shown in the org chart, a new duty on documentation is created and managed by one person. Bob compiles everything into the SOP book, but he uses everybody else's knowledge to do this. Other people often submit written pass downs to him. This SOP process is very informal compared with that of the larger Navy.

The simple structure accommodates the micro involvement that Luke must have with this people. When a customer is hacked, Luke must understand the details of all that took place so that he can properly speak with higher ups about the incident. The old hierarchy of middle management was an impediment to the business process of reporting incidents accurately and concisely up the chain of command.

Top management, Luke, still makes many decisions. The relevant input from personnel is still taken but ultimately the boss of the simple organization makes the final decisions.

After SAAM is applied, there are no organizational misfits.

As a technology, SAAM is no longer divisible from the organization. It is part of the entire network. Even though there may have been a SAAM specialist during the installation phase, SAAM is now pertinent to all network personnel. SAAM affects everybody.

The network with SAAM is now an adequate tool for the job at hand. Within the Center, applications that require real time delivery now function properly, all the time. The product that the Center delivers to the outside customers is a service. Now that SAAM is used on the entire path from the LAN inside the center, through the Internet, the

service that the Center provides is more real time. The Center now advertises that it delivers real time security service, with information continuance. Information continuance is the business term applied to the overall effect of active networks with QoS. [Aberdeen] Just as sure as the Internet will survive a nuclear blast on any node, the information coming from the Center will continue.

Technically, implementing information continuance is done when a network application is configured with service level agreements. High priority is given to specific flows within intrusion detection applications, remote monitoring applications, streaming video apps, voice over IP, and the teleconferencing app.

K. RATIONAL SYSTEMS MODEL

The rational systems model was used to analyze the organizational effects of SAAM, in attempt to view the Center as a system, with inputs, some business processing within, and some outputs.

In the Environment of the Center is the larger Navy, and specifically the CNO's office who has a large stake at seeing the Center succeed. The Internet Service Providers (ISP's) have a steak in the center once SAAM is enabled because these two organizations must have Service Level Agreements that cross-organizational boundaries. More than a contract, this can be viewed as almost a partnership for high quality telecom service. The customers are the Naval commands with networks, who use the special security services of the Center, and they obviously have a stake in the success of the Center.

There are opportunities in the environment. In order to remain a first rate Navy, the Center must provide an active network with QoS over IP capability, which SAAM

does. There will be a short term cost savings to show, but ultimately the Center must have an up to date network or cease to exist because the industry will over run it.

The threats in the environment include the hackers, both foreign and domestic. The Center has a short history of only five years. This provides opportunity to make a dramatic cultural change toward IT teamwork of professionals in uniform. This culture in IT fields is an ideal of the Navy but not actually supported by the current Navy, due to the dysfunctional rewards system and other factors. There is hostility in the environment. The rapid change of technology is a constant threat of making the Center obsolete.

Moving into the organization, the strategic view driving the Center is Network Centric Warfare (NCW). The objectives for today are to become a real time organization, able to run core Observe Orient Decide Act (OODA) loops in microseconds. The short-term path of attaining this strategy is to defend Naval networks.

There are many critical success factors to the successful adoption of SAAM and subsequent organizational improvement. There must be executive support for SAAM. Luke must drive it forward. No other position has the proper authority to make it a success. Even if someone higher than Luke forced the Center into using a new technology, it will be less successful than having Luke drive it forward. SAAM must be readily available in the ISP, as stated in one of the main assumptions. In today's terms, SAAM should be a feature bundled into Cisco IOS. The only viable business plan for SAAM right now is to get bought out by Cisco. No ISP would change all their routers to a small third part company just to gain SAAM. One note on the migration path must be made. SAAM will work with other QoS systems, such as RSVP and DiffServ, so as ISP's adopt the best QoS system, interoperability will be there to take advantage of legacy

systems. Furthermore, it is assumed that SAAM is a secure system, not introducing new gaping security threats to an already insecure Internet. Another critical success factor is the professional personnel inputs into the Center, each person IT literate with a subspecialty. Also, each person must be a team worker.

The following discussion of the internal works of the Center is organized into three different levels, the organization, groups, and people.

The size category has a small tactical organization of 10 people. The groups are formed around problems. Teamwork is required in almost every solution. The people are highly educated in IT, able to think abstractly on what technology can do to solve concrete problems, bringing credentials from universities and industry training certifications.

The tasks inside the organization vary from day to day, but the focused mission today for the organization is Computer Network Defense (CND). The groups are formed dynamically around problems. The subspecialty of each person is a unique contribution to the whole to support the leader, Luke, in making decisions.

The core technology is CND. For groups, red teaming new systems and ideas is a core technology and how the Center operates. These red teams are stressful tests. The day-to-day operations between people is a workflow process. Individuals cannot really complete a business process from end to end. People in the Center need each other.

The information and communications is really the industry in which the Center works. So this permeates the entire organization. Open communications are required between groups. Fast response is needed from each individual. E-mail and other network technologies are good examples of the speed of business on the Internet.

The structure of the Center has changed dramatically from a machine bureaucracy to an informal structure of informal roles or billets. Now the organization is able to evolve quickly around threats and opportunities. The new simple structure is like an adhocracy. The groups are again loose structures, dissolved after that project is complete. One permanent change due to SAAM is the merge between the telecom IP engineer roles. The structure relies on professionalized individuals in the IT field, which a broad knowledge domain or schema in IT. Only after one knows how to use a hammer will swinging it harder make forward progress. Hard work from such professionals will drive the organization forward.

The rewards system within the Center is based on results. The goal of the rewards system is teamwork and a customer focus. One reward, which is particular to the IT field, is "just for me training." The organization is willing to spend money on training classes or certifications as a reward to an individual IT career, even if that skill is not something that the Center needs immediately. Not only does this make people feel taken care of and motivate sharp people to continue working hard, but it provides more opportunity for the center to randomly find a new technology, a new opportunity for the organization. Individual rewards also help to make the goals of the individual in line with the organization. Standard Navy awards are given, representing intangible glory in the form of medals. Honor is a big driver for military people. This part of Navy culture should be maintained even though other cultural aspects are dramatically different. Command opportunity can be built at the center through a series of good written evaluations and fitness reports, specifically mentioning command in the future for this individual. The civil sector is able to reward its personnel with stock options. If that could legally be

worked into the Navy, this reward would further the Center as a rival in the private sector. The idea that 'nobody gets rich in the Navy' is a cultural barrier that would have to be overcome to offer stock options.

The flow of these rewards must be noted. The Center cannot really provide command opportunity for people, since the Center already has a commander. But the written evaluations from the Center can go a long way toward an individual's future command opportunities. The Navy boards actually do the prompting and command assignments. The Center can even leverage its high visibility and frequent contact with the CNO's office, by asking for the CNO's signature on the best awards and written appraisals.

The training in the organization takes into account the professional nature of the individuals, with a 75% rate of higher education. Training focuses this strong mental capacity to imagine a solution. There are routine group training events, about once a week in the Center. Some specific training occurs when new systems are installed. Individuals are trained toward industry certification. This might even be part of the Naval boards of review in the Center. Continuing education is a part of maintaining the schema for IT. Individuals go out in town to take a university class regularly.

The people influence in the Center is noted from the high visibility of the organization. Social facilitation theory applies to the organization, groups, and individuals. [Greenberg] The individuals in the Center have the potential for big impact on the Center, as well as the external Navy enterprise. Each person is a big fish in a big pond.

The symbolic/Interactive cultural aspects of the organization are items such as operating in Internet time. If the Internet moves in dog years, with a 7:1 ration, then the Center has to move fast. Three laws drive this rapid pace. Moore's Law, the tendency for the number of transistors on a CPU to double every 18 months, can be mapped to the tendency for the life cycle of a software system to live for only 18 months. Sidgemore's law notes the 16-fold increase in bits on the Internet every year, driving the need for QoS. Davidow's law is the need to obsolesce your own product before your competitor makes your product obsolete for you. [Lewis] The individuals in the Center understand all of these laws. This helps them remember that the Center is a fast twitch organization.

The outputs from the Center with SAAM include end-to-end information assurance. Automated network monitoring is improved with the smart agents programmed in. The groups produce a high quality, high group morale, which extends out from the Center to customers working with the center. The team of IT people at the customer end of the Center have a very positive experience of working with the Center. High customer satisfaction is a product of the Center. High individual satisfaction is a product of the Center. The individual member in the Center must be challenged and satisfied to continue the rapid pace of work required, noting that the rewards system and individual training reinforce this.

The overall outcome is that the Center provides unbeatable computer network defense service, rivaling even the private Internet companies if not beating them. The outstanding customer feedback arrives back to the Center immediately and reinforces good work. This feedback is also noted by the Naval superior commands, especially the

CNO. The final outcome is continued funding from the CNO. As the popular military saying goes, "If you just do this, you get to keep your job."

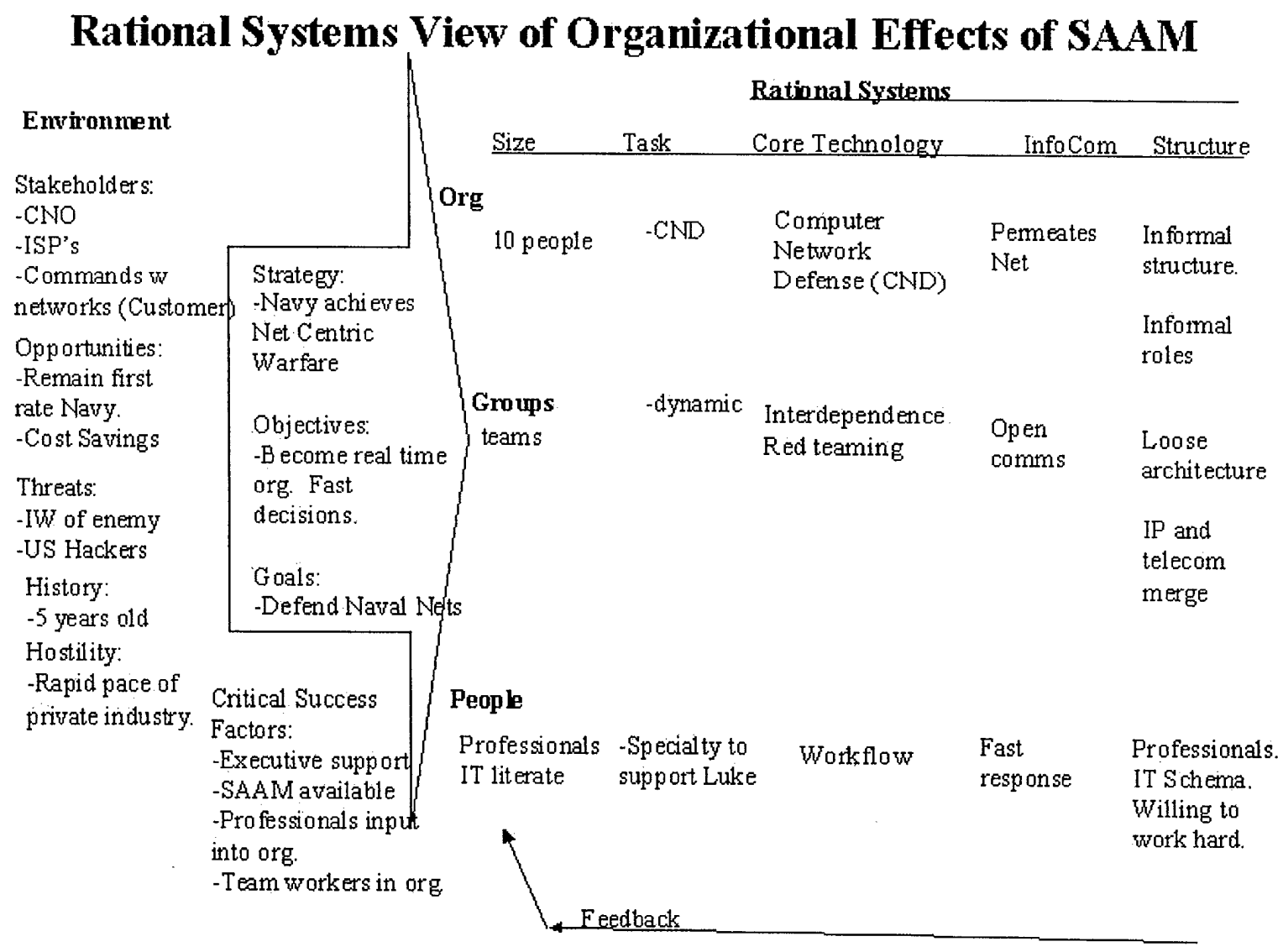


Figure 36. Rational Systems Model - page 1 of 2

Rational Systems View of Organizational Effects of SAAM

Human Resources- Career Mgt.

	Reward Systems	Flow	Training	People Influence	Symbolic/Interactive Formal/Culture	Outputs	Outcomes
Org	-Results based -Training -Awards -Command opportunity -Stock options -R.S. builds teamwork and cust. Svc.	-org writes fitreps, but External Navy promotes. -best awards get CNO signature.	-Professional 75% grad. Educated. -Routine everyday. -Some specific systems	-Hi. Vis. -social facilitation.	-Operate in Internet time. -Moore's Law. -Sidgemore's Law. -Davidow's Law. -Fast twitch org.	-Information assurance end to end. -Automate netmon -hi quality. -hi group morale. -hi individual satisfaction. -Hi cust. Satisfaction.	-Unbeatable CND Svc. -outstanding cutomer feedback. -continued funding from CNO.
Groups							
People	-goals of individual align w org.		-Schema for IT -allows for chunking. Fast learners. -Certification -continuing education	-Enterprise influence of individual.			

Feedback to start at "size."

Figure 37. Rational Systems Model - page 2 of 2

L. KEY LEARNING POINTS

Migrating from a legacy Local Area Network (LAN) to an active network has many organizational impacts.

1. Time Compression

The active network compresses the response time of intrusion detection systems, ultimately producing a faster service from the Center. On the Internet, faster is better. This time compression benefit facilitates risk-embracing leadership, which is needed to outpace the enemy. SAAM allows the Center, and Luke, to do near real time situation assessment and respond appropriately. SAAM makes the center into a fast twitch organization.

2. Information Parsed into Knowledge

Humans are still needed for problem solving. SAAM brings all the relevant data and reports to these decisions. For example, Luke could request reports of number of network probes that have found suspicious activity, analyzed in both space and time. These reports can be updated with real time data, using the QoS nature of SAAM.

3. IP and Telecom Barrier Removed

The barrier between the IP engineer billet and telecom billet has completely disappeared. This could become an organizational source of friction. Cooperation is easier to accomplish between individuals than would be if two separate divisions performed these two functions.

4. Teamwork

The SAAM network connects teams rather than individuals together.

5. Training Shipped to People

Training is done over streaming video and video teleconferencing calls. The travel budget is reduced by several orders of magnitude. Also, Video Teleconference VTC can replace the site visits from the Center. Training is shipped to the people, not the other way around.

6. Quality of Voice Over IP

It is a common misconception that the quality of a voice conversation over a dedicated phone line cannot be matched by using a regular IP network. This idea does not stand true in this case study. Quality of service does apply very well to services on IP networks using SAAM. The quality of a switched network is matched with SAAM over IP. The old QoS protocols failed for several reasons. The main reason is the service model is used, causing moments of congestion in the stream of data, and audible silences and repeats. The flow based routing model in SAAM eliminates bottlenecks from end to end. Once a flow is established, there is no competition from lower priority flows.

A related argument is that Voice over IP was already in place before SAAM came to be, suggesting that SAAM is not needed. In the late 1990's, companies such as Cisco had already successfully deployed Voice over IP worldwide, without using SAAM. The confounding factor is that Cisco owned the entire WAN used for VoIP. Most organizations rely on the public Internet, including the Navy. SAAM works over

heterogeneous networks. Many ISP's are in involved in a SAAM data flow from one SAAM endpoint to the other.

7. Fewer Layers in the Organization

The rigid command bureaucracy in the Naval culture is adjusted inside the Center to become more like an Internet company as found in Silicon Valley. As the network advances, the organization becomes more of a simple organization. Advanced networks tend to flatten out organizations.

M. CASE STUDY CONCLUSION

SAAM will affect the organization of the Center, a small tactical ten person simple organization. The LAN was a legacy system before SAAM was applied. SAAM decreased the phone bill by about an order of magnitude. Existing applications on the IP network became better. The Voice over IP system sounded like it was using a dedicated phone line through the switched circuits in the telephone company. Operational applications were enhanced, and now have information continuance. With the help of SAAM, the Center has become a fast twitch organization. Teams coagulated around solving problems in novel ways. Leadership is better able to assess the situation and respond in real time, if the active network has not responded already.

APPENDIX B. AUTHENTICATION PRIMER

A. CRYPTOGRAPHY

Cryptography is the science of protecting data by encryption. Cryptographic algorithms mathematically combine input plain text data and an encryption key to generate encrypted data, called cipher text. With a good cryptographic algorithm, it is computationally infeasible to reverse the encryption process and derive the plaintext data starting with only the cipher text. A decryption key is needed to perform the transformation.

In traditional, secret (or symmetric) key cryptography, the encryption and decryption keys are identical and thus share sensitive data. Parties wishing to communicate with secret-key cryptography must securely exchange their encryption/decryption keys before they can exchange encrypted data.

In contrast, the fundamental property of Public Key (PK) cryptography is that the encryption and decryption keys are different. Encryption with a public key encryption key is a "one-way" function. Plaintext turns into cipher text easily but the encryption key is irrelevant to the decryption process. A different decryption key (related but not identical to the encryption key) is needed to turn the cipher text back into plaintext. Thus, for PK cryptography every user has a pair of keys consisting of a public key and a private key. By making the public key available, it is possible for you to enable others to send you encrypted data that can only be decrypted using your private key. Similarly, you can transform data using your private key such that others can verify it originated from you. This latter capability is the basis for the digital signatures.

The separation between public and private keys in PK cryptography has allowed the creation of a number of new technologies. The most important of these are digital signatures, distributed authentication, secret key agreement via public key, and bulk data encryption without prior shared secrets.

There are a number of well-known PK cryptographic algorithms. Some, such as RSA (Rivest-Shamir-Adleman) and ECC (Elliptic Curve Cryptography), are general purpose in the sense they can support all of the above operations. Others support only a subset of these capabilities. Some examples include the Digital Signature Algorithm (DSA, which is a part of the U.S. government's Digital Signature's Standard), which is useful only for digital signatures, and Diffie-Hellman (D-H), which is used for secret key agreement.

The principle uses of PK cryptography and their operations can be summarized as case scenarios involving two imaginary clients, Bob and Alice. Assume that Bob and Alice can exchange information but do not have any prearranged shared secrets between them. The utilization of a Digital Signature (based on a mathematical transform that combines the private key with the data to be "signed" to such that:

-Only someone possessing the private key could have created the digital signature

-Anyone with access to the corresponding public key can verify the digital signature

-Any modification of the signed data (even changing only a single bit in a large file) invalidates the digital signature.

Digital signatures are themselves just data, so they can be transported along with the signed data they are intended to "protect". For example, Bob can create a signed e-

mail message to Alice and send the signature along with the message text, providing Alice the information required to verify the message origin. In addition, digital signatures provide a way to verify that data has not been tampered with (either accidentally or intentionally) while in transit from the source to the destination. Because of this, they can be exploited to provide a very high-assurance data integrity mechanism.

B. AUTHENTICATION

Public Key cryptography can be used to provide robust distributed authentication. Entity authentication guarantees that the sender of the data is the entity that the receiver thinks it is. One possible method involves the receiver, who is Alice in this example, sending a challenge to the sender, who is Bob in this example, encrypted with Bob's public key. Bob then decodes this challenge with his private key and sends it back to Alice, proving that he has access to the private key associates with the public key used to encrypt the challenge. An alternative is for Alice to send a plaintext challenge to Bob. Bob then combines the challenge with other information, which is digitally signed. Alice then uses Bob's public key to verify the signature and prove that Bob has the associated private key. The challenge makes this message unique and prevents replay attacks by a hostile third party. In either case, this is known as a "proof of possession" protocol because the sender is sender is proving that he has access to a particular private key.

C. SECRET KEY AGREEMENT

Another feature of PK cryptography is that it permits two parties to agree on a shared secret using public, and non-secure, communication networks. Basically, Bob and

Alice each generate a random number that will form half of the shared secret key. Bob then sends his half of the secret to Alice encrypted using her public key while Alice sends her half to Bob encrypted with his public key. Each side can then decrypt the message received from the other party, extract the half of the shared secret she did not generate herself, and combine the two halves to create the shared secret. Once the protocol is completed, the shared secret can be used for securing other communications.

Another model to distribute secret key pairs is to use a trusted third party, such as a Kerberos Server. The Key Distribution Center (KDC) in Kerberos is a trusted host, and distributes session key pairs to two computers that want to talk with each other over an untrusted network.

Given the plethora of security schemes available, a decision on which is the best one for SAAM must be made.

APPENDIX C. SELECTING AN AUTHENTICATION SYSTEM

A decision must be made concerning the best authentication scheme to work with SAAM. In this chapter, several authentication schemes are explored. Analytical tools were employed to help filter through the wide range of authentication schemes that may apply. These tools included a decision matrix and a more complex expert system called Logical Decisions for Windows. The recommendation to kerberize RSVP is also relevant to the decision and is discussed in this chapter.

A. DECISION MATRIX

The simple matrix was the first tool we attempted. The matrix allowed us to organize the different authentication systems as a literature research tool, summarizing what we thought we needed to know. The different authentication schemes are listed horizontally across the top and are defined as follows:

MITL - Man In The Loop

Kerberos - Kerberos authentication system.

PGP - Pretty Good Privacy

CA - Certificate Authority

IPSEC - IPsec

The attributes in question are listed in the left hand column, as shown below.

	MITL	Kerberos	PGP	CA	IPSEC
Usability	1	9	5	7	9
Performance	6	9	3	5	9
Suitability	1	9	2	7	9
Monetary Cost	1	8	9	5	8
Interoperability	1	9	7	8	9
Total	10	44	26	32	44

Table 7. Simple Matrix

The measures are listed in order of relative importance to SAAM. Each box receives a rank, on a scale from 1 to 10. These number assignments are arbitrarily set, based on our findings described below for each block. The bottom row shows the collective totals for each authentication scheme.

B. DESCRIPTION OF MEASURES

The following exploration and research outlines the evidence gathered to support the findings of fact. The thesis work is designed to provide a comprehensive authentication mechanism that will meet the following criteria for acceptance.

1. Usability

The ability for the investigated authentication scheme to meet "ease-of-use" functionality as well as the ability for the authentication scheme to appear "transparent" to the SAAM scheme of operations.

2. Overhead on Routers/Servers

The ability for the investigated authentication scheme to provide a small "foot print" in areas of hardware and bandwidth resources required for its implementation.

3. Suitability

The ability for the investigated authentication scheme to match the environment in which the SAAM architecture is to be deployed.

4. Monetary Cost

The ability for the investigated authentication scheme to provide an effective authentication measure of assurance while not requiring a large monetary cost over-head or a re-occurring cost over a significant period of time, thus producing a negative cost over the life of the SAAM design, implementation, a continued service.

5. Interoperability

The ability for the investigated authentication scheme to be incorporated into the fabric of the SAAM architecture without a significant measure of performance loss in its efficiency or effectiveness providing authentication to the SAAM Enterprise. This is inter-SAAM operability. The ability for the investigated authentication scheme to hosts outside of the SAAM fabric, i.e. an RSVP request sent into the SAAM fabric, this is to ensure that SAAM does not exclude cross platform compatibility with current routers. This is intra-SAAM operability.

C. DESCRIPTION OF AUTHENTICATION SCHEMES

1. "MITL" Man In The Loop

Man in the loop (MITL) security uses a human being. An individual is responsible for conducting all security issues, including the transport of Recognition Keys throughout the SAAM Active Site. An analysis of the capabilities for the MITL process to have effectiveness within the SAAM enterprise when matched against the five evaluation criteria has been provided in the following paragraphs.

a. MITL Usability

The MITL agent would be required for the interaction and introduction of new cipher keys into the Server-to-Server authentication distribution as well as to introduce the keys into the router-to-router authentication. The MITL agent does not provide for ease of use and is not a transparent agent that can be easily modified.

b. MITL Overhead on Routers/Servers

The MITL agent would require a user interface that is consistent across the SAAM Enterprise in order to interact with SAAM authentication mechanism. The designing and user interface ease of use would require additional programming support during the initial development. Allocating resources for the GUI would constitute distribution of a resource that could otherwise be dedicated to the sole purpose of speeding up the authentication process or routing. The Java Virtual Machine could encounter a "frozen" thread during the processing of the table of distribution keys and therefore cause the entire SAAM system to become unpredictable.

c. MITL Suitability

In the SAAM environment there will be numerous Servers interacting with each other throughout the SAAM enterprise.

Within each SAAM regional area of responsibility there can be as many as forty or more SAAM routers interacting with each other. The MITL representative would be required to visit each and every device to insert the latest Recognition Key table for that period. This would be too time consuming. The expense of man-hours would outweigh the benefits received.

d. MITL Monetary Cost

The actual cost in dollars for the inclusion of the authentication in the SAAM system will vary dependent upon the selected authentication mechanism. A total software based solution would demonstrate a large up front cost but would avoid a redistributing cost that could escalate if it were an MITL agent. Cost savings cannot be realized with the MITL agent since it is a human factor and that the human factor cost is ever increasing in the world of technology.

e. MITL Interoperability

The MITL represents a physical human being interacting within the authentication process, to install the unique keys to be implemented within the Server-to-Server communications and the unique keys to be utilized in support of the regional SAAM routers. These keys must also have a time assignment for its value of eligibility within the SAAM Enterprise.

If the MITL were to chosen, a guarded courier would have to deliver the new Recognition keys.

The dependency of an independent entity such as a MITL agent would mean that the entire authentication for the entire SAAM infrastructure would hinge on the MITL's ability to execute his/her duties with clockwork precision throughout the SAAM enterprise.

2. KERBEROS

The Kerberos is a distributed authentication service that allows a client running on behalf of a principle (user) to provide its identity to a verifier (server) without sending data across the network that might allow an attacker or the verifier to subsequently impersonate the principal. Kerberos optionally provides integrity and confidentiality for data sent between the client and server. Kerberos was developed in the mid-80's as part of MIT's Project Athena, originally designed to prevent spoofed logins. Students could log onto servers across the network and avoid password intercept vulnerabilities. This protection is ideal for SAAM. Kerberos must be integrated with other parts of the SAAM system. It does not protect all messages sent between two computers; it only protects the messages from software that has been written to use it. The Kerberos authentication scheme uses a series of encrypted messages to prove to a verifier that a client is running on behalf of a particular process. The Kerberos protocol is based in part on the Needham and Schroeder authentication protocol, but with changes to support the needs of the environment for which it was developed. Among these changes are the use of timestamps to reduce the number of messages needed for basic authentication, the addition of a "ticket-granting" service to support subsequent authentication without re-entry of a service's password, and different approaches to cross-realm authentication (authentication

of a principal registered with a different authentication server than the verifier). This is a simplified overview introduction to the Kerberos protocol; the paragraph by no means merits an all-exclusive overview of the Kerberos protocol and its volume of intricacies.

a. Kerberos Usability

The Kerberos measure of usability is measured by the ease of its use within the SAAM enterprise. Factors that require measurement to provide justification include the User Interface, and the ability to generate key tables and avoid redundancy of keys.

b. Kerberos Overhead on Routers/Servers

The level of overhead on SAAM routers is dependent on the measurement of the following metrics

Time of Key Generation and Validation

Processing of non-authenticated request

Service processing requirements

Level of difficulty in the coding process

Extra memory needed to support authentication

c. Kerberos Suitability

Kerberos serving as an integral agent can be designed and engineered to work exclusively within the SAAM environment as a service. The Service can be incorporated as a new authentication module for ease of maintenance and modification should the needs and requirements change. In particular the lifespan of the key may change. Also, another measurement of suitability is the ability for the Kerberos agent to draw information from higher and parallel agents within the Enterprise. For example, the

SAAM router can connect the SAAM Server and also the designated Key Distribution Center (KDC) router simultaneously.

d. Kerberos Monetary Cost

The monetary cost incurred by the SAAM project can be measured via the following metrics:

New Hardware required for authentication

The additional man-hours to implement authentication

e. Kerberos Interoperability

The purpose of Kerberos is to provide authentication within the SAAM Enterprise of Server-to-Server communications as well as provide authentication uniquely amongst all member routers of an autonomous SAAM router region. The SAAM Enterprise can have an internal process agent designed into its core as its authentication regulator or the Kerberos key distribution system can be a third party application used by the SAAM infrastructure. Designing Kerberos into the SAAM Network will provide for a stream lined automated set of tools that can be leveraged to provide robust authentication.

3. PGP

a. PGP Usability

PGP is relatively usable because it can be coded to run transparently before any messages are sent. However, PGP is slow because of the public key nature. It is not widely used in the commercial world.

b. PGP Overhead on Routers/Servers

PGP is highly CPU intensive due to the public key calculations.

c. PGP Suitability

PGP is not really suitable to distribute keys within SAAM and change the key tables periodically, because of the number of keys required. Also, there is no central key table in PGP. Each process needs a key. There are cases in SAAM where two or more processes are running. For example, the regional SAAM server is a child to the parent Server. One key is needed for each process. The number of keys needed would be [n(n-1)]/2 with n processes. PGP is not scalable.

d. PGP Monetary Cost

PGP is free, concerning the acquisition cost only. For any Information System, acquisition cost is really only about one third of the entire life cycle cost. The technical support and user training components are not free. Cost is about the same for all the alternatives except for MITL.

e. PGP Interoperability

PGP can allow an edge router in SAAM to talk to other systems. For example, PGP can be used to authenticate an RSVP request for using the SAAM fabric. PGP is freeware and open source. Another router could be coded to use PGP also, but PGP is not a popular solution today for router authentication because of the key distribution and management problems involved.

4. CERTIFICATE AUTHORITY (CA)

a. CA Usability

A certificate authority is a scalable key distribution system, and a possible authentication solution for SAAM. The centralized key table makes for easier key management and distribution than the PGP web of trust model. The trust model is the problem, not the technology itself. A router can be coded to call the CA transparently before any messages are sent. A CA is slow because of the public key nature. However, the server-based architecture offloads some of the CPU burden from the routers.

b. CA Overhead on Routers/Servers

Public key creation is highly CPU intensive. This is only relevant when granting a certificate. Four messages must pass over the network with a CA system. [Stallings]

c. CA Suitability

A CA is not completely suitable to distribute keys within SAAM. It would be difficult to refresh the Recognition Keys periodically. The advantages of a CA include centralized key management and a hierarchical nature of KDC's, which allow the CA to span all of SAAM fabric. This maps very well to SAAM hierarchy.

d. CA Monetary Cost

Third party vendors exist, such as Verisign, which charges $10 per certificate. This expense could work with this thesis research and may even work for the real SAAM system. It is likely that an in house SAAM CA would be less expensive in the long run. The monetary cost is just about the same for all alternatives except for MITL.

e. CA Interoperability

A CA can allow an edge router in SAAM to talk to other systems. For example, a CA can be used to authenticate an RSVP request for using the SAAM fabric. Any user that uses the same CA will be interoperable with SAAM.

5. IP SECURITY

IPsec is a layer three protocol that addresses the fundamental lack of security in the current Internet. IP next generation (IPng) was an exercise in the IETF to address the requirements for the next generation Internet. IPv6 was one of the resulting protocols, along with Classless Inter-Domain Routing (CIDR). IPv4 is the common internetworking protocol widely used on the Internet today. The security protocol has been extracted from IPv6 and can be used with IPv4. This security protocol with IPv4 is called IP Security (IPsec). IPsec uses an open standards architecture, allowing the specific encryption algorithm modules to be changed as CPU's become faster and faster.

IPsec requires public keys to operate, just as PGP does. A CA is an infrastructure to provide that public key. For this reason, CA does not compare well with public key technologies, such as PGP and IPsec.

a. IPsec Usability

Most routers on the market today now offer full support for IPsec. Configuration is said to be fairly simple. Likewise, IPsec could be coded into SAAM to be easy for the administrator to use.

b. IPsec Overhead on Routers/Servers

Many administrators use IPsec for virtual private networks (VPN's) to authenticate and encrypt all data traffic. IPsec was designed to run at layer 3, to encrypt all traffic. It may be possible to encrypt specific traffic associated with a network socket conversation, which would be application layer where SAAM runs. This is significant because SAAM only needs to authenticate signaling traffic, which runs at the application layer. Not all SAAM data traffic must be authenticated, and to do so would incur useless overhead.

c. IPsec Suitability

IPsec is fairly suitable because it is an open standard from the IETF, and it inherits all the benefits of operating at layer 3. That is to say, IPsec scales well, runs quickly, and without the knowledge of the administrator after it is configured to run. No manual intervention is needed after setup.

d. IPsec Monetary cost

As open source software, the ANSI C source code for IPsec can be free. Since there are several versions of encryption algorithms to choose from, companies will usually bundle IPsec as another service in a larger package, and charge money for the whole package. For example, Cisco IOS contains many different feature sets. Some versions of IOS contain IPsec capabilities.

e. IPsec Interoperability

Because it is an open standard, IPsec is interoperable. There is potential for some miscommunication between different devices. For example, it could be that IPsec in Microsoft windows 2000 might use a subtly different hash function in the

authentication header than the Cisco authentication header, rendering these two machines incompatible. It depends on how the standard is implemented.

6. Matrix Results

As you can see, the totals in the bottom of the matrix place Kerberos and IPsec in a tie for the best-fit authentication scheme. Further analysis was needed to differentiate between these two options. We chose to model our decision using decision support software.

F. LOGICAL DESCISIONS FOR WINDOWS

Logical Decisions for Windows (LDW) was used to assist in making this decision. LDW is a decision support system (DSS), which allows us to construct our decision model to consider all aspects relevant to the decision. LDW is based on utility theory. Provided that we setup the proper criteria with proper weights, the alternative with the highest overall utility is the best system. Goals are split into sub-goals, and logical measures are assigned to each sub goal. These metrics are all weighted appropriately. Trade-offs are made. The alternatives are applied to the model and data is finally entered for each category. This data entry is a little more quantitative than the matrix. LDW attempts to take qualitative knowledge and assign a quantity. The result is a recommendation from LDW for the best answer. The steps of logical decision analysis are: (1) structure the problem, (2) describe the alternatives, (3) review the preferences, and (4) rank alternatives and chose the best one.

1. Structure the Problem

a. Find the best authentication system

The main goal is to find the best authentication system. This main goal consists of several sub-goals, such as maximize usability, maximize performance, etc. The sub goals are shown in boxes on the following figure.

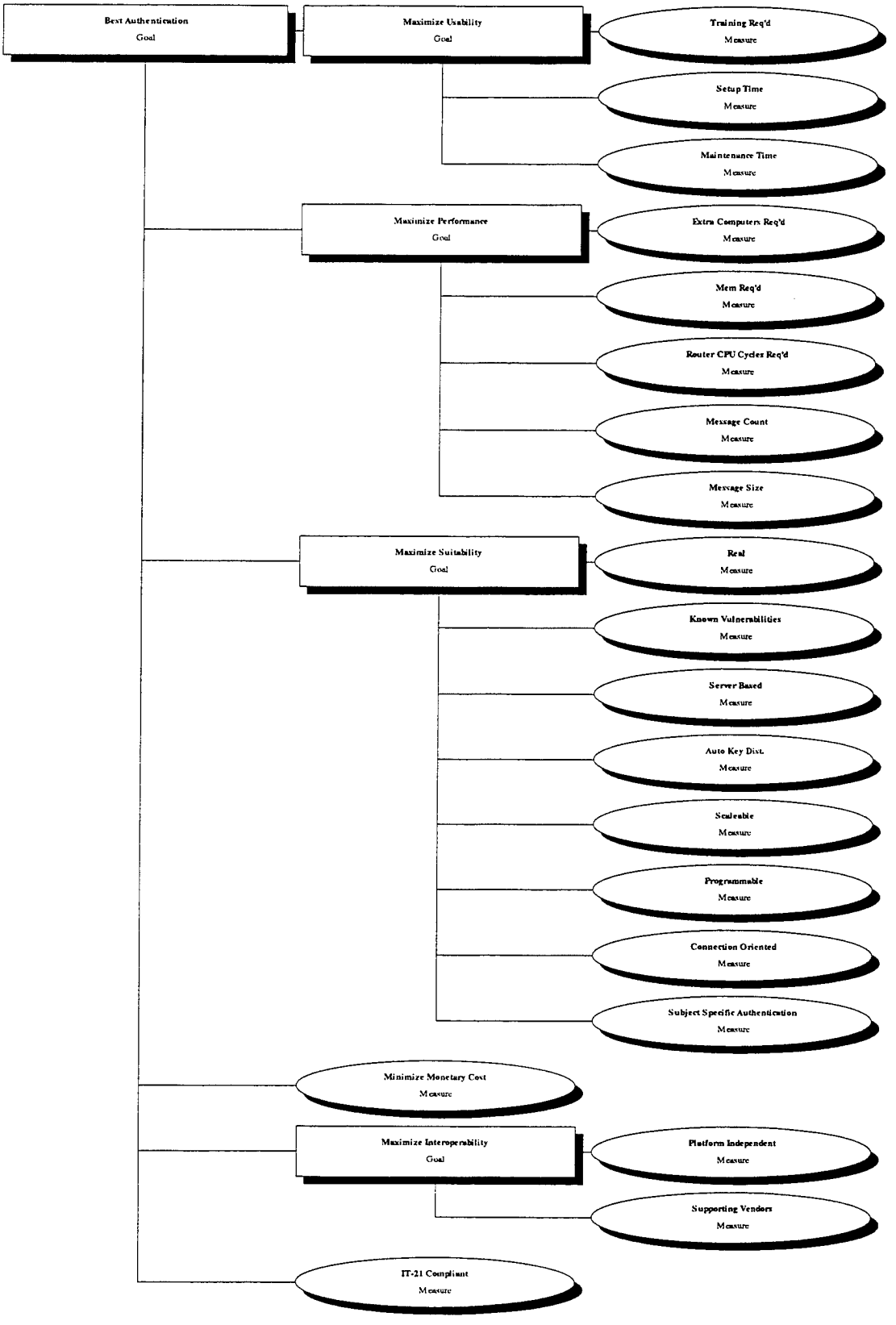


Figure 38. Goals and Sub Goals in LDW Model

Notice that a new sub goal was added into the LDW model. IT-21 compliance was considered to be important for SAAM. IT-21 compliance heavily influenced the final outcome of this model.

Comments were added to each sub-goal and measures, describing the purpose of the sub-goal or measure. The reports below capture our thoughts during run-time of our analysis, displayed in the form of comments.

a. Comments for Goals

Goal	Comment
Best Authentication Goal	
Maximize Usability Goal	Transparent to the user. How transparent to the user (administrator or IP Eng) is this Authentication mechanism?
Maximize Performance Goal	The least amount of overhead is (or the most performance, or efficiency) this a good authentication scheme.
Maximize Suitability Goal	Can this authentication mechanism run well with the SAAM system?
Maximize Interoperability Goal	Is this authentication scheme interoperable? 1. Within the SAAM system (inter-SAAM) 2. Outside of the SAAM system. For example, an RSVP request may be sent in from a foreign host.

Figure 39. Comments on Goals

b. Comments for Measures

Measure	Comment
Minimize Monetary Cost Measure	Cheapest price wins.
Training Requirement Measure	KISS principle. Don't want the administrator to spend too much time just learning the authentication method of SAAM.
Setup Time Measure	The system admin should be able to plug in the SAAM system and not have to worry about the authentication at all, ideally. Otherwise, he may turn it off.
Maintenance Time Measure	Ideally, once the authentication system is running, the system admin should not have to manually monitor it. It should run and fix itself

	without any problems.
Extra Computers Required Measure	Memory Requirements Measure. Better to have minimal memory requirements in the router. "If I have to buy extra RAM just for authentication that is bad." A core router's bottleneck is packet switching capacity. All else is peripheral. There is nothing particularly wrong with loading other tasks on the router except at the expense of core routing capacity. This is why offloading some tasks to a server is a benefit.
Router CPU Cycles Requirement Measure	Better to have minimal cpu cycles. For example, public key encryption calculations are usually more cpu intensive than one-way hash calculations. This concerns CPU cycles on Router only. IPsec relies on Router to generate session key, whereas Kerberos has KDC perform session key generation.
Message Count Measure	Better to have minimal number of messages sent during the handshake of authentication. This cuts down on bandwidth requirements.
Message Size Measure	Smaller authentication message size is better. This conserves bandwidth.
Real Measure	Is the authentication mechanism currently manufactured by vendors and sold today? If it is just an idea on paper, then it is not proven yet and very risky for us to try to implement. Is the authentication scheme currently used anywhere in the real world.
Known Vulnerabilities Measure	The Authentication Scheme should have the fewest known vulnerabilities possible.
Server Based Measure	Is the Authentication scheme server based? Can we use a central Key Distribution Center (KDC) with this scheme?
Auto Key Dist. Measure	The session key is automatically distributed.
Scaleable Measure	Can this authentication scheme scale from a prototype system with routers to the production Internet of millions?
Programmable Measure	Extensible. Can this authentication scheme be tailor fit into SAAM? That is, can this authentication scheme fit elegantly into the SAAM Architecture?
Connection Oriented Measure	Is the Authentication scheme connection oriented? That is to say, does this scheme allow A and B to authenticate while both are on line.

	If the authentication scheme allows A to authenticate while offline, then that is bad.
Platform Independent Measure	Can this authentication scheme run on our prototype in Java on WinNT and also run on Unix, Linux, Cisco IOS, or any other OS?
Supporting Vendors Measure	The more the better, at least Microsoft must support this authentication scheme. Other considerations are WUGS ATM vendor, Cisco, Red Hat and Sun.
Subject Specific Authentication Measure	Does the authentication scheme allow any application to run or is the authentication scheme specific to one subject running at the application layer? Assume that the resident agents in SAAM will require authentication each time a new resident agent is run.
IT-21 Compliant Measure	Is the authentication mechanism compliant with the general Navy?

Figure 40. Comments on Measures

2. Map Qualitative Goals to Quantitative Measures

The strength of LDW is that qualitative goals can be mapped to quantitative measures. For example, the goal to maximize usability contains issues such as training required, setup time, and maintenance time. Each of these measures can be quantified in units of hours. In cases where we still do not know how many hours a particular system will require, we set up units of high, medium and low, indicating over 100 hours, about 50, and less than 20 hours respectively, so that we could compare one system directly to another. Quantifying the data input into LDW to some degree overcomes the uncertainty of our inputs in the matrix used above.

Once we have a unit assigned to each measure, a utility is assigned to those units. A utility value of 1 is the best, while a utility value of 0 is the worst. Most utilities are a decimal value between zero and one.

The assessment summary report below shows what quantitative units were placed on each metric. This is how we get a number from a qualitative measure.

Note that LDW utilities that are assessed directly in this table (not MUF's) are shown in black and white. Data for the utilities with single measure utility functions (MUF's) are not shown in the table below. For an explanation of what a MUF is, please see the LDW documentation.

a. Common Units		
Auto Key Distribution	Label	Utility
	1. Yes	1
	0. No	0
Connection Oriented	1. Yes	1
	2. No	0.5
ExtraComputers Requirements	More than 4 platforms.	1
	2-4 platforms	0.5
	1 platform	0.1
IT-21 Compliant	1. Yes	1
	0. No	0
Maintenance Time	High	0
	Medium	0
	Low	0
Memory Required	most preferred - 0MB	1.000
	mid preferred -16MB	0.500
	least preferred -32MB	0
Programmable	1. Yes	1
	0. No	0
Real	1. Yes	1

	0. No	0
Router CPU Cycles Required	3.High	0.1
	2. Medium	0.25
	1. Low	1
Scaleable	High-100,000,000	1
	Medium-100,000	0.5
	Low - 100	0.01
Server Based	1. Yes	1
	0. No	0
Setup Time	Short Time	1
	Medium Time	0.5
	Long Time	0
Subject Specific Authentication	1. Yes	1
	0. No	0
Supporting Vendors	High5 or more	1
	Medium 2- 5	0.5
	Low 1	0.1
Training Requirements	Low	1
	Medium	0.5
	High	0

Figure 41. Assessment Summary Report

Continuous Single-Measure Utility Functions

Extra Computers Req'd

least preferred range	preferred level	most preferred utility	most preferred level	mid-preferred utility	mid-preferred level	preferred utility
0.000	3.000	0.000	0.000	1.000	1.000	0.250

Known Vulnerabilities

least preferred range	preferred level	most preferred utility	most preferred level	mid-preferred utility	mid-preferred level	preferred utility
0.000	2.000	0.500	0.000	1.000	1.000	0.900
1.000	5.000	0.000	2.000	0.500	3.500	0.100

Mem Req'd

least preferred range	preferred level	most preferred utility	most preferred level	mid-preferred utility	mid-preferred level	preferred utility
0.000	32.000	0.000	0.000	1.000	16.000	0.500

Message Count

least preferred range	preferred level	most preferred utility	most preferred level	mid-preferred utility	mid-preferred level	preferred utility
0.000	7.000	0.000	1.000	1.000	4.000	0.500

Message Size

least preferred range	preferred level	most preferred utility	most preferred level	mid-preferred utility	mid-preferred level	preferred utility
0.000	1500.000	0.000	1.000	1.000	750.500	0.500

Minimize Monetary Cost

least preferred range	preferred level	most preferred utility	most preferred level	mid-preferred utility	mid-preferred level	preferred utility
0.000	1000.000	0.000	0.000	1.000	500.000	0.500

Maximize Usability: a Utility function based on these tradeoffs:

	A	B
Setup Time (Hours)	40	0
Maintenance Time (Hrs/wk)	0	5

Figure 42. Assessment Summary of MUF's

Notice how we considered the scalable requirement as separate from server based. These two variables are independent because scalability can be achieved without necessarily using a server-based solution. SAAM requires scalability to deploy to the global Internet. The server-based requirement is nice to have to match the current implementation of SAAM.

The supporting vendors requirement considered only the number of vendors that support the technology. We did not think in terms of cross vendor interoperability, which really should be considered.

b. Weights
a. Best Authentication
A Utility function based on these tradeoffs:
A B
No interactions assessed
b. Maximize Interoperability
Utility function based on these tradeoffs:
A B
No interactions assessed
c. Maximize Performance
Utility function based on these tradeoffs:
A B
No interactions assessed
d. Maximize Suitability
Utility function based on these tradeoffs:
A B
No interactions assessed
e. Maximize Usability
Utility function based on these tradeoffs:
A B
Setup Time (Hours) 40 0
Maintenance Time (Hrs/wk) 0 5
No interactions assessed

Figure 43. Assessment Summary Report

Once the goals and sub goals are properly given measures with specific units, the next step to forming the decision model is to assign weights to the sub goals. Please see the tradeoff summary graph.

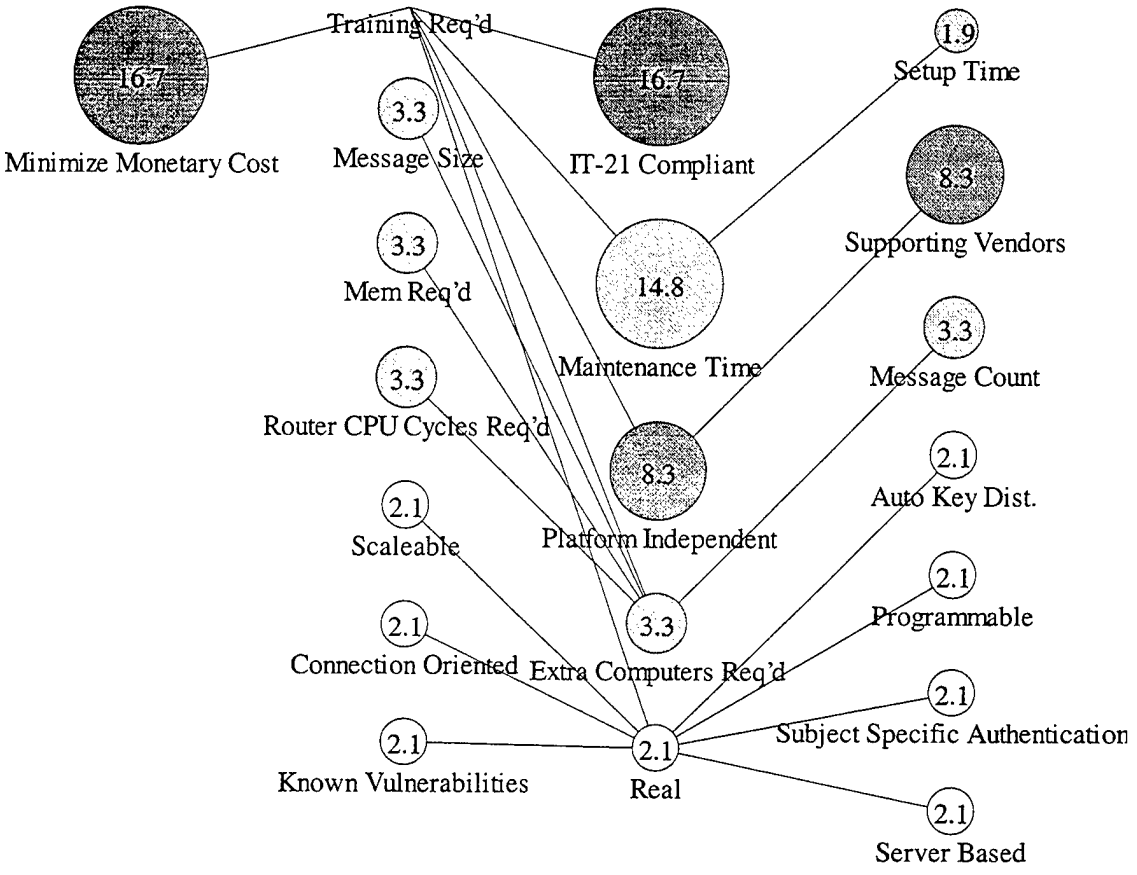


Figure 44. Tradeoff Summary Graph

The Tradeoff summary graph shows the percentage weights associated with each measure in the entire LDW model.

Note that a MUF is a Multi-measure Utility Function. In this model, each sub goal can be considered a MUF. In the figure above, weights in a common parent goal, looking up the tree of a sub goal branch, are the same color.

The same weights data is presented below in a report format.

Percentage Weights for Preference Set Authentication Mechanisms in SAAM

Measure	Percentage Weight	Effective Weight
Minimize Monetary Cost	16.7	16.7
Training Requirements	0.0	0.0
Setup Time	1.9	1.9
Maintenance Time	14.8	14.8
Extra Computers Req'd	3.3	3.3
Mem Req'd	3.3	3.3
Router CPU Cycles Req'd	3.3	3.3
Message Count	3.3	3.3
Message Size	3.3	3.3
Real	2.1	2.1
Known Vulnerabilities	2.1	2.1
Server Based	2.1	2.1
Auto Key Dist.	2.1	2.1
Scaleable	2.1	2.1
Programmable	2.1	2.1
Connection Oriented	2.1	2.1
Platform Independent	8.3	8.3
Supporting Vendors	8.3	8.3
Subject Specific Authentication	2.1	2.1
IT-21 Compliant	16.7	16.7

Figure 45. Weights Applied to Measures

3. Describe the alternatives

The next step is to determine what alternative systems should go into the model. We were not experts in the field of authentication and many systems were chosen based on general research on the Internet, and interviews with computer security experts. The

alternative systems chosen were then applied into the decision model we had constructed. The alternatives are listed below.

a. ***MITL Alternative: Man In The Loop***

b. ***Kerberos Alternative:***

c. ***PGP Alternative: Pretty Good Privacy***

d. ***CA Alternative: Certificate Authority***

e. ***IPsec Alternative:***

f. ***SSL Alternative: Secure Socket Layer***

g. ***S/MIME Alternative: Secure MIME***

4. Review the Preferences

The model now can be viewed as a large spreadsheet. Like the simple matrix example before this LDW discussion, we had to assign values to each box in this new spreadsheet. Again, the values assigned this time have undergone more scrutiny and structure. See the matrix view below.

ALTERNATIVES							
	Minimize Monetary Cost	Training Req'd	Setup Time	Maintenance Time	Extra Com-puters Req'd	Mem Req'd	Router CPU Cycles Req'd
MITL	1000	High	Long Time	High	2	0	1. Low
Ker-beros	0	Medium	Medium Time	Low	1	4	1. Low
PGP	0	Medium	Medium Time	High	0	4	3. High
CA	10	High	Long Time	High	1	4	2. Medium
IPsec	0	Medium	Medium Time	Low	1	4	2. Medium
SSL	0	Medium	Medium Time	High	0	4	3. High
S/MIME	0	Medium	Medium Time	High	0	4	3. High

	Message Count	Message Size	Real	Known Vulnerabilities	Server Based	Auto Key Dist.
MITL	1	100	1. Yes	3	0. No	0. No
Kerberos	4	100	1. Yes	0	1. Yes	1. Yes
PGP	2	100	1. Yes	2	0. No	0. No
CA	5	100	1. Yes	0	1. Yes	0. No
IPsec	3	100	1. Yes	0	0. No	1. Yes
SSL	1	100	1. Yes	1	0. No	0. No
S/MIME	1	100	1. Yes	1	0. No	0. No

	Scaleable	Programmable	Connection Oriented	Platform Independent	Supporting Vendors
MITL	Medium – 100,000	0. No	1. Yes	1platform	Low 1
Kerberos	High – 100,000,000	1. Yes	1. Yes	More than 4 platforms.	High 5 or more
PGP	Low - 100	1. Yes	2. No	More than 4 platforms.	High 5 or more
CA	High – 100,000,000	0. No	1. Yes	More than 4 platforms.	High 5 or more
IPsec	High –	1. Yes	1. Yes	More than 4	High 5 or

	100,000,000			platforms.	more
SSL	Medium – 100,000	1. Yes	1. Yes	More than 4 platforms.	High 5 or more
S/MIME	Medium – 100,000	1. Yes	2. No	More than 4 platforms.	High 5 or more

	Subject Specific Authentication	IT-21 Compliant
MITL	0. No	0. No
Kerberos	1. Yes	1. Yes
PGP	0. No	0. No
CA	1. Yes	0. No
IPsec	1. Yes	0. No
SSL	0. No	0. No
S/MIME	0. No	0. No

Figure 46. Data Entered into Model

5. Rank Alternatives and Chose the Best One

See the graphical result below, a final product of LDW. It becomes clear to use Kerberos for SAAM.

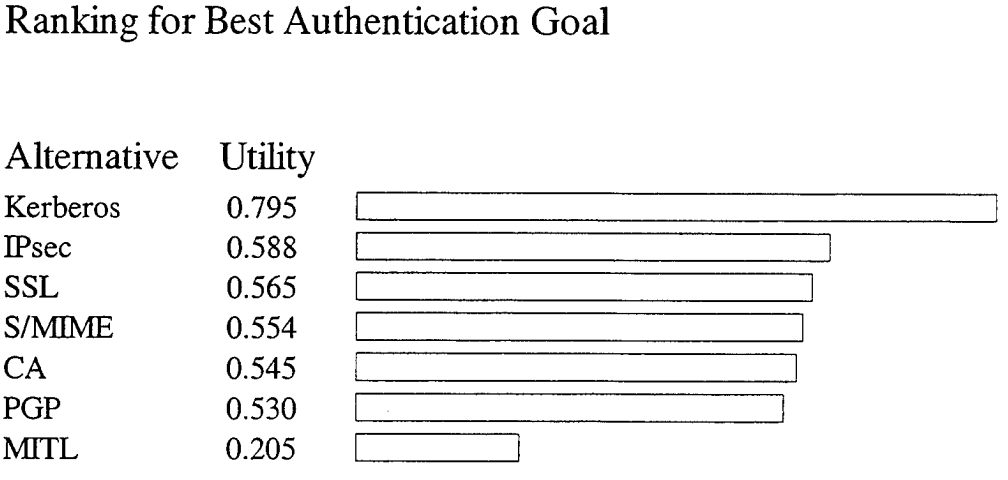


Figure 47. LDW Result - Ranking for Best Authentication

G. CONCURRENT RESULTS

The IETF addresses the same situation we found ourselves in. The question of using IPSEC or Kerberos for a distributed network protocol was explored in detail. We arrived at the same decision to use Kerberos. A quote from RFC2747, Section 1.2, follows.

> Why not use the Standard IPSEC Authentication Header? One obvious question is why, since there exists a standard authentication mechanism, IPSEC ..., we would choose not to use it.
>
> This was discussed at length in the working group, and the use of IPSEC was rejected for the following reasons. The security associations in IPSEC are based on destination address. It is not clear that RSVP messages are well defined for either source or destination based security associations, as a router must forward PATH and PATH TEAR messages using the same source address as the sender listed in the SENDER TEMPLATE. RSVP traffic may otherwise not follow exactly the same path as data traffic. Using either source or destination based associations would require opening a new security association among the routers for which a reservation traverses. In addition, it was noted that neighbor relationships between RSVP systems are not limited to those that face one another across a communication channel. RSVP relationships across non-RSVP clouds ... are not necessarily visible to the sending system. These arguments suggest the use of a key management strategy based on RSVP router to RSVP router associations instead of IPSEC. From Ref. [RFC2747, sec 1.2]

Our analysis developed as we learned more and more about IPsec and Kerberos. When we arrived at the LDW result to use Kerberos, we still had questions. However, it was reassuring to find that the IETF made the same decision as we did because RFC2747 has undergone some degree of peer review.

APPENDIX D. SAAM INTEGRATED PROTOTYPE CODE

A. SAAM CODE

1. Packet Factory

```
  /**
   * When instantiated to receive packets, the PacketFactory
   * thread waits until a SAAMPacket arrives, then it calls
   * the processPacket method.
   */
  public void run(){

System.out.println("I can call the scmSecurityManager from here"); //pjs
ScmSecMgr secMgr = new ScmSecMgr(false);

     while (true){
       synchronized (inputLock){
         if (inputQueue.isEmpty()){
           try{
             gui.sendText("Input packetfactory going to sleep...");
             inputFactoryAsleep = true;
             inputLock.wait();
             inputFactoryAsleep = false;
             gui.sendText("Input packetfactory resumed");
           }
```

2. scmSecMgr

```
//'Module:         Security Manager
//'Programmer:     Pete
//'Date:           Tuesday 4Jun00
//'Description:    THIS CODE WAS TRANSLATED FROM VB
```

```
//'          Code module, now a Java Class, to declare global variables
//'               and the main procedure, which loads
//'               the main security manager form
//'Make sure that the startup object is this module.

//*************************************
//*Name: scmSecMgr
//*Authors: pjs
//*Purpose: To handle all security functions for authenticaiton in SAAM.
//*************************************//

//do import statements here
//We need to create the security package.
package saam.security;
//import a whole bunch of stuff, just like the Translator.
import java.net.ServerSocket;
import java.net.SocketException;
import java.net.Socket;
import java.net.InetAddress;
import java.net.UnknownHostException;
import java.util.TooManyListenersException;
import java.util.Vector;
import java.util.Hashtable;
import java.io.IOException;
import java.io.OutputStream;
import java.io.InputStream;

import com.dstc.security.kerberos.Kerberos;
import com.dstc.security.kerberos.KerberosContext;
import com.dstc.security.kerberos.gssapi.*;
import java.security.*;
import java.io.*;
import GSSClient;
import GSSServer;

//import saam.message.*;
```

```
//import saam.router.*;
//import saam.util.SAAMRouterGui;
//import saam.util.Array;
//import saam.agent.ResidentAgent;
//import saam.message.MessageProcessor;
//import saam.event.*;
//import saam.net.*;
//import saam.control.*;

public class ScmSecMgr {
//**
//* Instance variables
//*
//* State Variables
//**/

  GSSClient client;
  GSSServer server;

//'Dimension the variables
//'Variables to store the selected scenario
/*
Public Const giBOOTSCENE        As Integer = 1
Public Const giNEWJOINSCENE     As Integer = 2
Public Const giKEYREFRESHSCENE  As Integer = 3
Public giScenario               As Integer
*/
  public static final int giBOOTSCENE      = 1,
              giNEWJOINSCENE    = 2,
              giKEYREFRESHSCENE = 3;
  public int        giScenario        = 1;

//'Variables to store which host I am
/*
Public Const giKDC      As Integer = 1
Public Const giServer   As Integer = 2
```

```
Public Const giRouterA  As Integer = 3
Public Const giRouterB  As Integer = 4
Public giWhoAmI         As Integer
*/
  public static final int giKDC      = 1,
              giServer = 2,
              giRouterA  = 3,
              giRouterB  = 4;
  public int        giWhoAmI   = 1;

//'Variables to store my security state
/*
Public Const giNEWNODE      As Integer = 1
Public Const giTRUSTED      As Integer = 2
Public Const giRECOGNIZED   As Integer = 3
Public Const giINVALID      As Integer = 4
Public giSecurityState      As Integer
*/
  public static final int giNEWNODE        = 1,
              giTRUSTED         = 2,
              giRECOGNIZED    = 3,
              giINVALID         = 4;
  public int        giSecurityState = 1;

//'Variables to store my message type
/*
Public Const giJOINREQUEST      As Integer = 1
Public Const giTRUSTSESSREQ     As Integer = 2
Public Const giTRUSTSESSRESP    As Integer = 3
Public Const giKTREQ            As Integer = 4
Public Const giKTRESP           As Integer = 5
Public Const giDCM              As Integer = 6
Public Const giUCM              As Integer = 7
*/
//'flow ID list in real SAAM will supercede this list here.
//Where in SAAM can I find the list of flow ID's?
```

```
//Public giWorkingMsg             As Integer
//Never was used.

//Public gstMsgContents       As String //'Ctrl Msg sent out to SAAM
public String gstMsgContents;

//Note.  The constructor must be placed physically before main().
public ScmSecMgr ( boolean wantClient ){

    // add DSTC to the list of crypto providers
    Provider dstc = new com.dstc.security.provider.DSTC();
    Security.addProvider(dstc);

  if (wantClient) {
    runClient();
  } else {
    runServer();
  }
/**
/*Functions performed are to:
/*1.  Determine my security state.
/*2.  Send the appropriate message out, according to the state.
/**/
  //NodeState = pNodeState;
  System.out.println( "From inside the scmSecMgr constructor." );
/*
  //hard code for now. pjs.
  nodeState = 3;

  //determine my security state.
  if ( nodeState == SecurityManager.NEWNODE ){
    System.out.println( " I am in a New Node State." );
  }
  if ( nodeState == SecurityManager.TRUSTED ){
    System.out.println( " I am in a Trusted State." );
```

```
    }
    if ( nodeState == SecurityManager.RECOGNIZED ){
      System.out.println( " I am in a Recognized State." );
    }
    if ( nodeState == SecurityManager.INVALID ){
      System.out.println( " I am in an Invalid State." );
    } //end if/thens

    //Send the appropriate msg out according to the state.
    if ( nodeState == SecurityManager.NEWNODE ){
      System.out.println( " I am sending a JoinRequest msg." );
      //new saam.message.JoinRequest( ).init( );
    }
    if ( nodeState == SecurityManager.TRUSTED ){
      System.out.println( " I am sending a KeyTableRequest msg." );
    }
    if ( nodeState == SecurityManager.RECOGNIZED ){
      System.out.println( " I am sending a KeyTableResponse msg." );
    }
    if ( nodeState == SecurityManager.INVALID ){
      System.out.println( " I am invalid.  I have no Node Secret and cannot join
SAAM." );
    }
*/

} //end scmSecMgr constructor

  /**
   * SecurityManager contains a main method to allow us to
   * run this class independent of the rest of SAAM temporarily
   * for diagnostics and testing only.  When this is finished, we//'ll take the
   * main method out of here and leave main only in the Translator.
   *
   * A main method to enable command-line instantiation.
   */
```

```
/*
Sub Main()
//'Declare variables to initialize default states
giScenario = 1
giWhoAmI = 1
giSecurityState = 1
//'Display the first form
Load frmSecurityManager
frmSecurityManager.Show vbModal
End Sub
*/

  public static void main( String args[] ) {

//giScenario = 1;
//Produces a compile error.
//Can't make a static reference to nonstatic variable giScenario in class
saam.security.scmSecMgr
//try putting these assignments on delcaration statements
// giWhoAmI = 1;
// giSecurityState = 1;

  // add DSTC to the list of crypto providers
  Provider dstc = new com.dstc.security.provider.DSTC();
  Security.addProvider(dstc);

  if (args.length != 1) {
    System.out.println( "Specify \"server\" or \"client\"" );
    return;
  }

  if (args[0].equalsIgnoreCase("server")) {
    System.out.println( "Running Server" );
    ScmSecMgr secMgr = new ScmSecMgr(false);
  }
  else {
```

```
      System.out.println( "Running Client" );
      ScmSecMgr secMgr = new ScmSecMgr(true);
    }

    System.out.println( "I am inside the main method" );
    System.out.println( "I was once VB but now am Java" );
    } //main

    public void runClient(){
      String server_host = "charlie.net1.cs.nps.navy.mil";
      int server_port = 4321;
      String service = "gssserver@charlie.net1.cs.nps.navy.mil";

      String userName = "gssclient";
      String userPass = "gssclient";

      String msg = "This is a test message";

      msg = msg + '\0';

      Kerberos kerb = Kerberos.getDefault();

      try
      {
        kerb.requestTGTWithPassword( userName, userPass );
        // change the username in the context to reflect the name for
        // which the tgt was requested.  Otherwise, the name shows the
        // name with which the user logged onto the local machine.
        KerberosContext context = kerb.getKerberosContext();
        context.setUsername(userName);

        Socket socket = new Socket(InetAddress.getByName(server_host),
server_port);
        BufferedOutputStream bos = new
BufferedOutputStream(socket.getOutputStream());
```

```
      DataInputStream dis = new DataInputStream(new
BufferedInputStream(socket.getInputStream()));

      GSSClient client = new GSSClient(dis, bos);

      client.establishContext(service);

      // display context flags
      client.displayStatus();

      client.sendMessage(msg);

    }
    catch (GSSException e){
      System.out.println("GSS-API error: " + e.getMessage());
    }
    catch (Exception e){
      System.out.println(e.getMessage());
    }
  }

  public void runServer(){
    int port = 4321;
    String service = "gssserver@charlie.net1.cs.nps.navy.mil";

    GSSServer server = new GSSServer(port, service);
    server.run();
  }

} //end class
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX E. WEB RECOGNITION KEY PROTOTYPE CODE

FILE: SAAMSERVERUPDATE.ASP

```
<%

  'DB CONFIGURATION CONSTANTS

  DIM DB_CONNECTIONSTRING

  'DB_CONNECTIONSTRING = "DRIVER={MICROSOFT ACCESS DRIVER (*.MDB)};DBQ=" &
SERVER.MAPPATH("./DB_SCRATCH.MDB") & ";"

  'CONST DB_USERNAME = "USERNAME"

  'CONST DB_PASSWORD = "PASSWORD"


  'UNCOMMENT THE ABOVE TO USE THIS WITH AN ACCESS DATABASE PROGRAM OR USE THE BELOW
WITH A SQL SERVER.

  'DB_CONNECTIONSTRING = APPLICATION("SQLCONNSTRING") & "UID=" &
APPLICATION("SQLUSERNAME") & ";PWD=" & APPLICATION("SQLPASSWORD") & ";"

  DB_CONNECTIONSTRING = "DRIVER=SQL
SERVER;SERVER=FILESERVER;UID=SA;PWD=SQLINTHE99;APP=MICROSOFT DEVELOPMENT
ENVIRONMENT;WSID=CIA2;DATABASE=SITESERVER3"

  DIM OBJRECORDSET

  SET OBJRECORDSET = SERVER.CREATEOBJECT("ADODB.RECORDSET")

  OBJRECORDSET.OPEN "SERVERKDC", DB_CONNECTIONSTRING, ADOPENKEYSET, ADLOCKPESSIMISTIC,
ADCMDTABLE

  OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER

%>

<HTML>

  <!-- #INCLUDE FILE="ADOVBS.INC" -->

<HEAD>

<META HTTP-EQUIV="CONTENT-TYPE" CONTENT="TEXT/HTML; CHARSET=WINDOWS-1252">
```

```
<META NAME="GENERATOR" CONTENT="MICROSOFT FRONTPAGE 4.0">
<META NAME="PROGID" CONTENT="FRONTPAGE.EDITOR.DOCUMENT">
<TITLE>DESTROY CIPHER PAGE</TITLE>
</HEAD>

<BODY BACKGROUND="IMAGES/BACKGROUND.GIF" TEXT="#800000">
<%
'*********************************
FUNCTION CLEARCIPHER()
  OBJRECORDSET.MOVEFIRST
  IF NOT OBJRECORDSET.EOF AND NOT OBJRECORDSET.BOF THEN
      OBJRECORDSET.DELETE
  END IF
  OBJRECORDSET.CLOSE
  SET OBJRECORDSET = NOTHING
END FUNCTION

'*****************************************************************
FUNCTION CLEARALLCIPHER()
  OBJRECORDSET.CLOSE
  OBJRECORDSET.OPEN "SERVERKDC DELETE FROM SERVERKDC", DB_CONNECTIONSTRING,
ADOPENKEYSET, ADLOCKPESSIMISTIC, ADCMDTABLE
  OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER
  OBJRECORDSET.CLOSE
  SET OBJRECORDSET = NOTHING
END FUNCTION

'*************************************************************
```

```
SELECT CASE REQUEST.FORM("OPT")
    CASE 0
    CLEARCIPHER()
    CASE 1
    CLEARALLCIPHER()
END SELECT
RESPONSE.REDIRECT("TOP_SAAMSERVER.ASP")
%>

</BODY>

</HTML>
```

FILE:TOPSAMSERVER.ASP

```
<%

  'DB CONFIGURATION CONSTANTS
   DIM DB_CONNECTIONSTRING
  'DB_CONNECTIONSTRING = "DRIVER={MICROSOFT ACCESS DRIVER (*.MDB)};DBQ=" &
SERVER.MAPPATH("./DB_SCRATCH.MDB") & ";"
  'CONST DB_USERNAME = "USERNAME"
  'CONST DB_PASSWORD = "PASSWORD"

  'UNCOMMENT THE ABOVE TO USE THIS WITH AN ACCESS DATABASE PROGRAM OR USE THE BELOW
WITH A SQL SERVER.
  'DB_CONNECTIONSTRING = APPLICATION("SQLCONNSTRING") & "UID=" &
APPLICATION("SQLUSERNAME") & ";PWD=" & APPLICATION("SQLPASSWORD") & ";"
```

```
   DB_CONNECTIONSTRING = "DRIVER=SQL
SERVER;SERVER=FILESERVER;UID=SA;PWD=SQLINTHE99;APP=MICROSOFT DEVELOPMENT
ENVIRONMENT;WSID=CIA2;DATABASE=SITESERVER3"
  DIM OBJRECORDSET
  SET OBJRECORDSET = SERVER.CREATEOBJECT("ADODB.RECORDSET")
  OBJRECORDSET.OPEN "SERVERKDC", DB_CONNECTIONSTRING, ADOPENKEYSET, ADLOCKPESSIMISTIC,
ADCMDTABLE
  OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER
%>
<HTML>
  <!-- #INCLUDE FILE="ADOVBS.INC" -->
<HEAD>
<META HTTP-EQUIV="CONTENT-TYPE" CONTENT="TEXT/HTML; CHARSET=WINDOWS-1252">
<META NAME="GENERATOR" CONTENT="MICROSOFT FRONTPAGE 4.0">
<META NAME="PROGID" CONTENT="FRONTPAGE.EDITOR.DOCUMENT">
<TITLE>SAAM AUTHENTICATION TECHNOLOGY PREVIEW</TITLE>
<META NAME="AUTHOR" LANG="EN" CONTENT="LUIS E. VELAZQUEZ">
<META NAME="REPLY-TO" HTTP-EQUIV="REPLY-TO" CONTENT="VMAN@NEWSGUY.COM">
<META NAME="DESCRIPTION" HTTP-EQUIV="DESCRIPTION" CONTENT="SAAM SERVER TECHNOLOGIES"
LANG="EN">
<META NAME="OWNER" CONTENT="VMAN@NEWSGUY.COM">
<META NAME="KEYWORDS" HTTP-EQUIV="KEYWORDS" CONTENT="SAAM SERVERS ROUTERS"
LANG="EN">
<META NAME="COPYRIGHT" CONTENT="LUIS E. VELAZQUEZ">
<META HTTP-EQUIV="REFRESH" CONTENT="60;
URL=HTTP://WWW.SQLNETWORKS.COM/THESIS/SIMULATION/TOP_SAAMSERVER.ASP">
<BASE TARGET="_SELF">
</HEAD>
<%
```

```
  SET FILEOBJECT = SERVER.CREATEOBJECT("SCRIPTING.FILESYSTEMOBJECT")

  DIR = REQUEST.SERVERVARIABLES("SCRIPT_NAME")

  DIR = STRREVERSE(DIR)

  DIR = MID(DIR, INSTR(1, DIR, "/"))

  DIR = STRREVERSE(DIR)

  HITSFILE = SERVER.MAPPATH(DIR) & "\ROUTER.TXT"


  ON ERROR RESUME NEXT

  SET INSTREAM= FILEOBJECT.OPENTEXTFILE (HITSFILE, 1, FALSE )

  OLDHITS = TRIM(INSTREAM.READLINE)

  NEWHITS = OLDHITS + 1

  SET OUTSTREAM= FILEOBJECT.CREATETEXTFILE (HITSFILE, TRUE)

  OUTSTREAM.WRITELINE(NEWHITS)

  L=LEN(NEWHITS)

  I = 1

  FOR I = I TO L

          NUM = MID(NEWHITS,I,1)

  DISPLAY = DISPLAY & "<IMG SRC=""/IMAGES/" & NUM & ".GIF""">"

  NEXT

%>



<BODY BGCOLOR="#000080" TEXT="#008000" BACKGROUND="IMAGES/BACKGROUND.GIF">


<P ALIGN="CENTER"><B>SAAM SERVER AND KDC</B></P>

<P ALIGN="CENTER"><% RESPONSE.WRITE DISPLAY %></P>


<%
```

```
'*********************************************

FUNCTION GENLETTER()

RANDOMIZE()

LETTERNUMBER = INT(25*RND)

SELECT CASE LETTERNUMBER


 CASE 0

  GENLETTER = "A"

 CASE 1

  GENLETTER = "B"

 CASE 2

  GENLETTER = "C"

 CASE 3

  GENLETTER = "D"

 CASE 4

  GENLETTER = "E"

 CASE 5

  GENLETTER = "F"

 CASE 6

  GENLETTER = "G"

 CASE 7

  GENLETTER = "H"

 CASE 8

  GENLETTER = "I"

 CASE 9

  GENLETTER = "J"

 CASE 10

  GENLETTER = "K"
```

```
CASE 11
 GENLETTER = "L"
CASE 12
 GENLETTER = "M"
CASE 13
 GENLETTER = "N"
CASE 14
 GENLETTER = "O"
CASE 15
 GENLETTER = "P"
CASE 16
 GENLETTER = "Q"
CASE 17
 GENLETTER = "R"
CASE 18
 GENLETTER = "S"
CASE 19
 GENLETTER = "T"
CASE 20
 GENLETTER = "U"
CASE 21
 GENLETTER = "V"
CASE 22
 GENLETTER = "W"
CASE 23
 GENLETTER = "X"
CASE 24
 GENLETTER = "Y"
```

```
 CASE 25

 GENLETTER = "Z"


END SELECT


END FUNCTION

'*********************************

FUNCTION GENNEWTICKET

RANDOMIZE()

'--- FORMAT FOR NEW KEYS IS AS FOLLOWS

'--- 1ST  DIGIT ALPHANUMERIC

'--- 2ND  DIGIT ALPHABETICAL

'--- 3RD  DIGIT NUMERIC

'--- 4TH  DIGIT NUMERIC

'--- 5TH  DIGIT NUMERIC

'--- 6TH  DIGIT ALPHABETICAL

'--- 7TH  DIGIT NUMERIC

'--- 8TH  DIGIT ALPHABETICAL

'--- 9TH  DIGIT NUMERIC

'--- 10TH DIGIT NUMERIC

'--- 11TH DIGIT NUMERIC

'--- 12TH DIGIT NUMERIC


KEYOPTION = INT(2*RND)

SELECT CASE KEYOPTION

    CASE 0
```

```
    GENNEWTICKET = CSTR(INT(9*RND)) & GENLETTER() & CSTR(INT(9*RND)) & CSTR(INT(9*RND)) &
CSTR(INT(9*RND)) & GENLETTER() & CSTR(INT(9*RND)) & GENLETTER() & CSTR(INT(9*RND)) &
CSTR(INT(9*RND)) & CSTR(INT(9*RND)) & CSTR(INT(9*RND))

    CASE 1

    GENNEWTICKET =     GENLETTER() & GENLETTER() & CSTR(INT(9*RND)) & CSTR(INT(9*RND)) &
CSTR(INT(9*RND)) & GENLETTER() & CSTR(INT(9*RND)) & GENLETTER() & CSTR(INT(9*RND)) &
CSTR(INT(9*RND)) & CSTR(INT(9*RND)) & CSTR(INT(9*RND))

END SELECT


END FUNCTION

'*********************************

FUNCTION ADDNEWTICKETS(X)

  DIM  TICKETNUMBER

  RANDOMIZE()

  SESSION("NEWTICKETOKAY") = TRUE

  TICKETNUMBER = CSTR(GENNEWTICKET())

  OBJRECORDSET.ADDNEW

  'OBJRECORDSET.FIELDS("CIPHERTEXT") = CSTR(TICKETNUMBER)

  OBJRECORDSET.FIELDS("CIPHERKEY") = TICKETNUMBER

  OBJRECORDSET.UPDATE

END FUNCTION

'*********************************

FUNCTION UPDATEACTIVECIPHER


  OBJRECORDSET.CLOSE

  OBJRECORDSET.OPEN "SERVERKDC WHERE ACTIVE=1 AND EXPIRED=0", DB_CONNECTIONSTRING,
ADOPENKEYSET, ADLOCKPESSIMISTIC, ADCMDTABLE

  OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER

  'RESPONSE.WRITE OBJRECORDSET.FIELDS("CIPHERKEY")
```

```
    OBJRECORDSET.FIELDS("ACTIVE") = 0
    OBJRECORDSET.FIELDS("EXPIRED") = 1
    OBJRECORDSET.UPDATE
    OBJRECORDSET.CLOSE
    OBJRECORDSET.OPEN "SERVERKDC WHERE ACTIVE=0 AND EXPIRED=0", DB_CONNECTIONSTRING,
ADOPENKEYSET, ADLOCKPESSIMISTIC, ADCMDTABLE
    OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER
    OBJRECORDSET.FIELDS("ACTIVE") = 1
    OBJRECORDSET.FIELDS("EXPIRED") = 0
    OBJRECORDSET.UPDATE
    OBJRECORDSET.CLOSE
    OBJRECORDSET.OPEN "SERVERKDC", DB_CONNECTIONSTRING, ADOPENKEYSET, ADLOCKPESSIMISTIC,
ADCMDTABLE
    OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER
END FUNCTION
'*********************************
FUNCTION GETACTIVECIPHER


    OBJRECORDSET.CLOSE
    OBJRECORDSET.OPEN "SERVERKDC WHERE ACTIVE=1", DB_CONNECTIONSTRING, ADOPENKEYSET,
ADLOCKPESSIMISTIC, ADCMDTABLE
    OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER
    RESPONSE.WRITE OBJRECORDSET.FIELDS("CIPHERKEY")
    OBJRECORDSET.CLOSE
    OBJRECORDSET.OPEN "SERVERKDC", DB_CONNECTIONSTRING, ADOPENKEYSET, ADLOCKPESSIMISTIC,
ADCMDTABLE
    OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER
```

```
END FUNCTION



'*****************************************************************

'*****************************************************************

'* VARIABLES

'* X  | HOW MANY TICKETS TO GENERATE

'* KK | COUNTER TO INCREMENT AS KEYS ARE GENERATED

'* YY | HOW MANY RECORDS WERE IN THE DATABASE PRIOR TO ACTUAL GENERATION

 IRECORDCOUNT = OBJRECORDSET.RECORDCOUNT

  DIM X, KK, YY, IRECORDCOUNT

            X  = 0

            KK = 100

'*****************************************************************

  IF IRECORDCOUNT = 0 THEN

            'SESSION("NEWTICKETOKAY") = FALSE

            WHILE  IRECORDCOUNT < KK

                        RESPONSE.WRITE "CREATING CIPHER TEXT FOR THE MASTER SERVER VIA THE KDC"
%><BR><%

                        RESPONSE.WRITE "TOTAL KDC RECORD COUNT" & IRECORDCOUNT %><BR> <%

                        ADDNEWTICKETS(X)

                        RESPONSE.WRITE "RECORD ADDED VERIFYING INTEGRITY" %><BR><%

                        IRECORDCOUNT= IRECORDCOUNT + 1

            WEND

            OBJRECORDSET.FIELDS("ACTIVE") = 1

            OBJRECORDSET.UPDATE

            RESPONSE.WRITE "CIPHER KEY ACTIVATED"

  END IF
```

```
'*************************************************************

' LOOP THROUGH RECORDSET AND DISPLAY RESULTS
   OBJRECORDSET.MOVEFIRST
   IF NOT OBJRECORDSET.EOF THEN
            UPDATEACTIVECIPHER()
    OBJRECORDSET.MOVEFIRST

    %>
   <P ALIGN="CENTER"> <FORM ACTION="RIGHTPANE.ASP" METHOD="GET">
    <SELECT NAME="ID" SIZE="1">
    <OPTION><%= GETACTIVECIPHER() %></OPTION>
    <%
    ' CONTINUE UNTIL WE GET TO THE END OF THE RECORDSET.
    DO WHILE NOT OBJRECORDSET.EOF
    %>
  <OPTION VALUE="<%= OBJRECORDSET.FIELDS("ID") %>"><%= OBJRECORDSET.FIELDS("CIPHERKEY")
& ", " & OBJRECORDSET.FIELDS("EXPIRED") & ", " & OBJRECORDSET.FIELDS("ACTIVE") %></OPTION>

      <%
      ' GET NEXT RECORD
      OBJRECORDSET.MOVENEXT
   LOOP
    %>
    </SELECT>
    <P ALIGN="CENTER"><I><B> SAAM. 2 MASTER ROUTER KDC UPDATE</B></I></P>
    </FORM>
    <%
```

```
    END IF

OBJRECORDSET.CLOSE
%>

<!--WEBBOT BOT="GENERATEDSCRIPT" PREVIEW=" " STARTSPAN --><SCRIPT
LANGUAGE="JAVASCRIPT"><!--
FUNCTION FRONTPAGE_FORM2_VALIDATOR(THEFORM)
{

 IF (THEFORM.OPT.SELECTEDINDEX < 0)
 {
  ALERT("PLEASE SELECT ONE OF THE \"CIPHER DELETE OPTION\" OPTIONS.");
  THEFORM.OPT.FOCUS();
  RETURN (FALSE);
 }
 RETURN (TRUE);
}
//--></SCRIPT><!--WEBBOT BOT="GENERATEDSCRIPT" ENDSPAN --><FORM METHOD="POST"
ACTION="TOP_SAAMSERVERUPDATE.ASP" NAME="FRONTPAGE_FORM2" ONSUBMIT="RETURN
FRONTPAGE_FORM2_VALIDATOR(THIS)">
 <P ALIGN="CENTER"><INPUT TYPE="SUBMIT" VALUE="SUBMIT" NAME="B1"><B><!--WEBBOT
 BOT="VALIDATION" S-DISPLAY-NAME="CIPHER DELETE OPTION" B-VALUE-REQUIRED="TRUE"
 --><SELECT SIZE="1" NAME="OPT">
  <OPTION VALUE="0">DELETE SINGLE CIPHER</OPTION>
```

<OPTION VALUE="1">DELETE ALL CIPHERS</OPTION>

</SELECT>UPDATE

MASTER CIPHER</B></P>

</FORM>

</BODY>

</HTML>

FILE:TOP_SAAMSERVERUPDATE.ASP

<%

'DB CONFIGURATION CONSTANTS

DIM DB_CONNECTIONSTRING

'DB_CONNECTIONSTRING = "DRIVER={MICROSOFT ACCESS DRIVER (*.MDB)};DBQ=" &
SERVER.MAPPATH("./DB_SCRATCH.MDB") & ";"

'CONST DB_USERNAME = "USERNAME"

'CONST DB_PASSWORD = "PASSWORD"

'UNCOMMENT THE ABOVE TO USE THIS WITH AN ACCESS DATABASE PROGRAM OR USE THE BELOW
WITH A SQL SERVER.

'DB_CONNECTIONSTRING = APPLICATION("SQLCONNSTRING") & "UID=" &
APPLICATION("SQLUSERNAME") & ";PWD=" & APPLICATION("SQLPASSWORD") & ";"

DB_CONNECTIONSTRING = "DRIVER=SQL
SERVER;SERVER=FILESERVER;UID=SA;PWD=SQLINTHE99;APP=MICROSOFT DEVELOPMENT
ENVIRONMENT;WSID=CIA2;DATABASE=SITESERVER3"

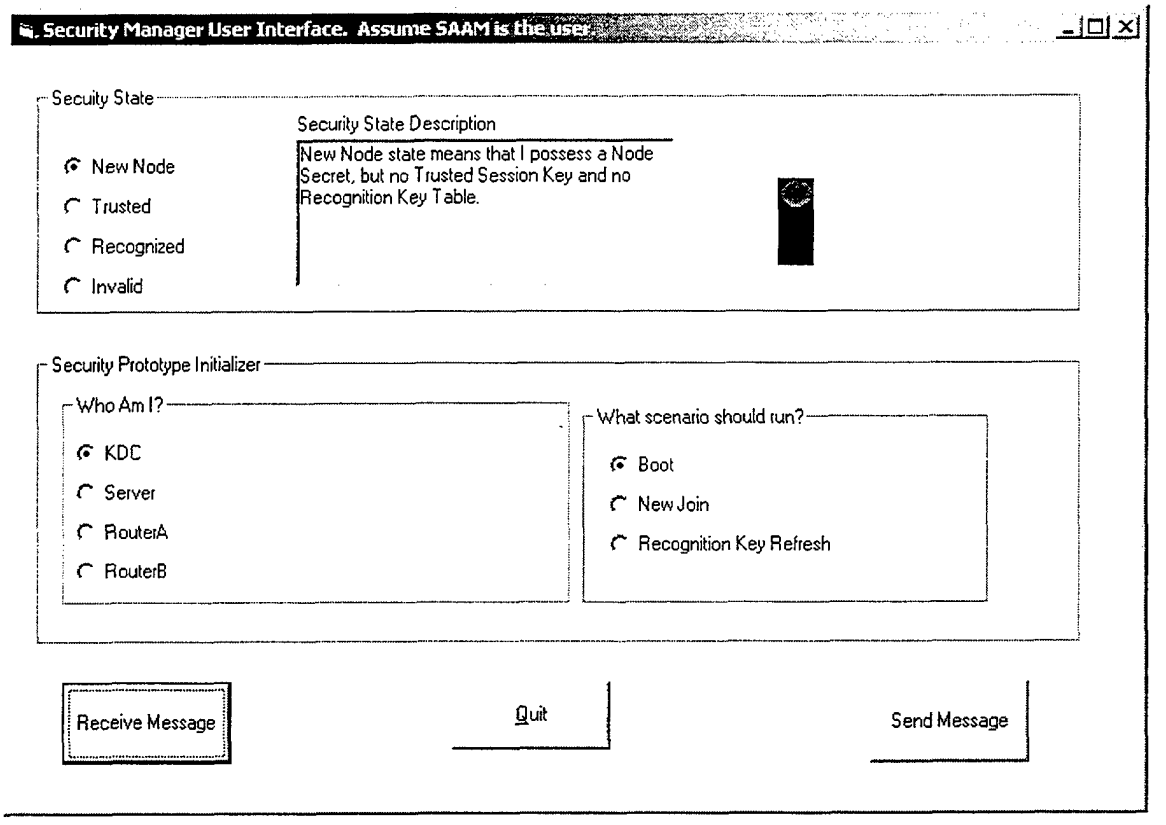
DIM OBJRECORDSET

```
  SET OBJRECORDSET = SERVER.CREATEOBJECT("ADODB.RECORDSET")
  OBJRECORDSET.OPEN "SERVERKDC", DB_CONNECTIONSTRING, ADOPENKEYSET, ADLOCKPESSIMISTIC,
ADCMDTABLE
  OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER
%>
<HTML>
  <!-- #INCLUDE FILE="ADOVBS.INC" -->
<HEAD>
<META HTTP-EQUIV="CONTENT-TYPE" CONTENT="TEXT/HTML; CHARSET=WINDOWS-1252">
<META NAME="GENERATOR" CONTENT="MICROSOFT FRONTPAGE 4.0">
<META NAME="PROGID" CONTENT="FRONTPAGE.EDITOR.DOCUMENT">
<TITLE>DESTROY CIPHER PAGE</TITLE>
</HEAD>

<BODY BACKGROUND="IMAGES/BACKGROUND.GIF" TEXT="#800000">
<%
'*********************************
FUNCTION CLEARCIPHER()
  OBJRECORDSET.MOVEFIRST
  IF NOT OBJRECORDSET.EOF AND NOT OBJRECORDSET.BOF THEN
          OBJRECORDSET.DELETE
  END IF
  OBJRECORDSET.CLOSE
  SET OBJRECORDSET = NOTHING
END FUNCTION

'************************************************************
FUNCTION CLEARALLCIPHER()
```

```
    OBJRECORDSET.CLOSE
    OBJRECORDSET.OPEN "SERVERKDC DELETE FROM SERVERKDC", DB_CONNECTIONSTRING,
ADOPENKEYSET, ADLOCKPESSIMISTIC, ADCMDTABLE
    OBJRECORDSET.CACHESIZE = 15 ' CUTS DOWN ON ROUND TRIPS TO OUR SQL SERVER
    OBJRECORDSET.CLOSE
    SET OBJRECORDSET = NOTHING
END FUNCTION

'*************************************************************

SELECT CASE REQUEST.FORM("OPT")
    CASE 0
    CLEARCIPHER()
    CASE 1
    CLEARALLCIPHER()
END SELECT
RESPONSE.REDIRECT("TOP_SAAMSERVER.ASP")
%>

</BODY>

</HTML>
```

APPENDIX F. VISUAL BASIC PROTOTYPE CODE

A. FRMSECURITYMANAGER




```
Private Sub cmdQuit_Click()

End


End Sub


Private Sub cmdRcvMsg_Click()

' go to the Process message form
```

```
Me.Hide
frmStatusFrame.Show vbModless
frmRcvMsg.Show vbModal
End Sub

Private Sub cmdSendMsg_Click()

'Update the Status frame with the selections made by the user, SAAM himself
If optBoot.Value = True Then
   giScenario = giBOOTSCENE
   'Debug lines added pjs.
   'Print giBOOTSCENE
   'Print giScenario
ElseIf optNewJoin.Value = True Then
   giScenario = giNEWJOINSCENE
ElseIf optRecKeyRefresh.Value = True Then
   giScenario = giKEYREFRESHSCENE
Else
   MsgBox "Please make scenaio selection", vbOKOnly, "Oops"
End If

' go to the Choose Message form
```

```
Me.Hide

frmStatusFrame.Show vbModless

frmChooseMsg.Show vbModal

End Sub

Private Sub Form_Activate()

'Try to force a refresh of that status frame.

'frmStatusFrame.Hide

End Sub

Private Sub optBoot_Click()

'Store that we will run the SAAM network boot scenario

giScenario = giBOOTSCENE

End Sub

Private Sub optInvalid_Click()

'Describe invalid state

giSecurityState = giINVALID
```

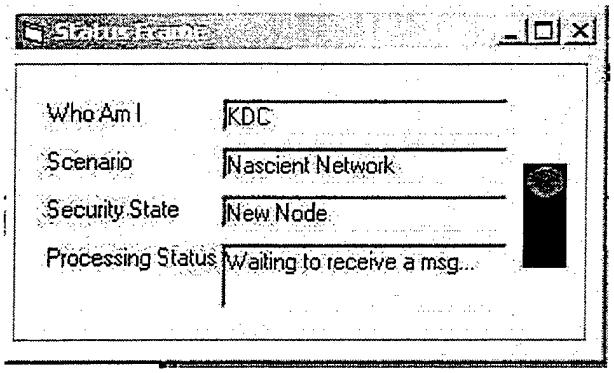
```
lblSecStDescription.Caption = "Invalid state means that my Node Secret has been
revoked from the KDC.  If the enemy has compromised me, they will not be able to use
me for much longer."
imgRed.Visible = False
imgYellow.Visible = False
imgGreen.Visible = False
imgInvalid.Visible = True
End Sub

Private Sub optKDC_Click()
'store that who i am is the KDC

giWhoAmI = giKDC

End Sub

Private Sub optNewJoin_Click()
'store that we will run the new join scenario

giScenario = giNEWJOINSCENE
End Sub

Private Sub optNewNode_Click()
```

```
'Store the security state, Describe the New Node State, display appropriat image.

giSecurityState = giNEWNODE
lblSecStDescription.Caption = "New Node state means that I possess a Node Secret, but
no Trusted Session Key and no Recognition Key Table."
imgRed.Visible = True
imgYellow.Visible = False
imgGreen.Visible = False
imgInvalid.Visible = False
End Sub

Private Sub optRecKeyRefresh_Click()
'store that we will run the recognition key refresh scenario

giScenario = giKEYREFRESHSCENE
End Sub

Private Sub optRecognized_Click()
'Describe Recognized Node State

giSecurityState = giRECOGNIZED
lblSecStDescription.Caption = "Recogninzed state means that I possess a valid
Recognition Key and I can complete authentication in SAAM now."
```

```
imgRed.Visible = False
imgYellow.Visible = False
imgGreen.Visible = True
imgInvalid.Visible = False
End Sub

Private Sub optRouterA_Click()
'store that i am routerA

giWhoAmI = giRouterA
End Sub

Private Sub optRouterB_Click()
'store that I am routerB

giWhoAmI = giRouterB
End Sub

Private Sub optServer_Click()
'store that i am the saam server

giWhoAmI = giServer
End Sub
```

```
Private Sub optTrusted_Click()
'Describe the Trusted State

giSecurityState = giTRUSTED
lblSecStDescription.Caption = "Trusted state means that I a valid Trusted Session Key,
but no Recognition Key Table."
imgRed.Visible = False
imgYellow.Visible = True
imgGreen.Visible = False
imgInvalid.Visible = False
End Sub
```

B. FRMSTATUSFRAME




```
Private Sub Form_Activate()


'Get and set the values of whoami and all the other output labels
'lblWhoAmI_Val.Caption = giWhoAmI
If giWhoAmI = 1 Then
   lblWhoAmI_Val.Caption = "KDC"
   ElseIf giWhoAmI = 2 Then
   lblWhoAmI_Val.Caption = "Server"
   ElseIf giWhoAmI = 3 Then
   lblWhoAmI_Val.Caption = "RouterA"
   ElseIf giWhoAmI = 4 Then
   lblWhoAmI_Val.Caption = "RouterB"
   Else: MsgBox "I do not know who I am", vbExclamation, "Oops"
End If
```

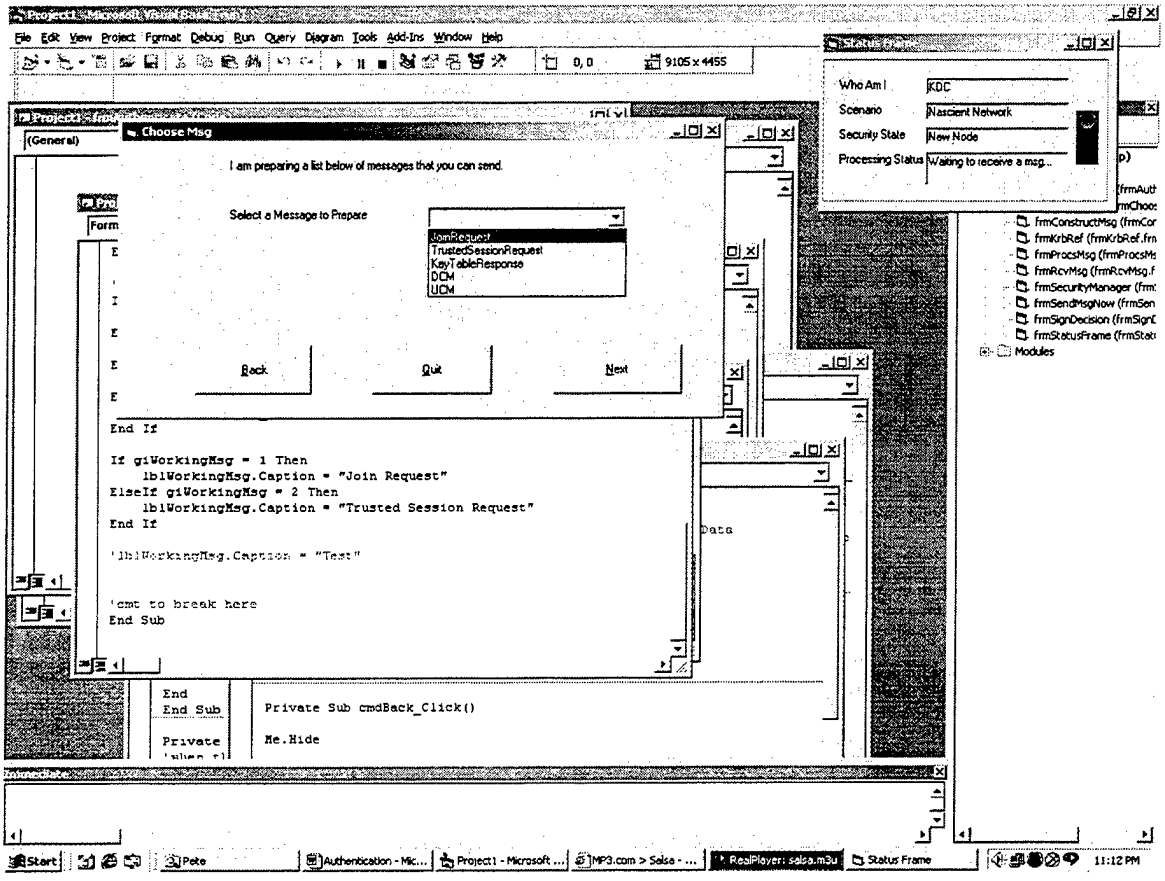
```
If giSecurityState = 1 Then
   lblSecState_Val.Caption = "New Node"
   'set traffic light
   imgRed.Visible = True
   imgYellow.Visible = False
   imgGreen.Visible = False
   imgInvalid.Visible = False

   ElseIf giSecurityState = 2 Then
   lblSecState_Val.Caption = "Trusted"
   imgRed.Visible = False
   imgYellow.Visible = True
   imgGreen.Visible = False
   imgInvalid.Visible = False

   ElseIf giSecurityState = 3 Then
   lblSecState_Val.Caption = "Recognized"
   imgRed.Visible = False
   imgYellow.Visible = False
   imgGreen.Visible = True
   imgInvalid.Visible = False
```

```
            ElseIf giSecurityState = 4 Then
            lblSecState_Val.Caption = "Invalid"
            imgRed.Visible = False
            imgYellow.Visible = False
            imgGreen.Visible = False
            imgInvalid.Visible = True

            Else: MsgBox "I do not know what my security state is", vbExclamation,
"Oops"
        End If

        If giScenario = 1 Then
            lblScenario_Val.Caption = "Nascient Network"
            ElseIf giScenario = 2 Then
            lblScenario_Val.Caption = "New Join"
            ElseIf giScenario = 3 Then
            lblScenario_Val.Caption = "Recognition Key Refresh"
            Else: MsgBox "I do not know what scenario to run", vbExclamation, "Oops"
        End If

        'Give a few processing status messages for common situations
        If giWhoAmI = giRouterB And giSecurityState = giNEWNODE And giScenario
= giNEWJOINSCENE Then
```

```
            lblProcsStatus_Val.Caption = "I just got a DCM.  Preparing my Join Request
msg..."
        ElseIf giWhoAmI = giRouterA And giSecurityState = giNEWNODE And
giScenario = giNEWJOINSCENE Then
            lblProcsStatus_Val.Caption = "Waiting for a Join Request..."
        ElseIf giWhoAmI = giServer And giSecurityState = giNEWNODE And
giScenario = giNEWJOINSCENE Then
            lblProcsStatus_Val.Caption = "Waiting for signed UCM's"
        ElseIf giWhoAmI = giKDC And giSecurityState = giNEWNODE And giScenario
= giNEWJOINSCENE Then
            lblProcsStatus_Val.Caption = "Waiting for Trusted Session Request msg..."
        End If

        If giWorkingMsg = 1 Then
            lblWorkingMsg.Caption = "Join Request"
        ElseIf giWorkingMsg = 2 Then
            lblWorkingMsg.Caption = "Trusted Session Request"
        End If

        'lblWorkingMsg.Caption = "Test"

        'cmt to break here
```

End Sub

C. FRMCHOOSEMESSAGE



Private Sub cboMsgList_Change()

'frmStatusFrame.lblWorkingMsg.Caption = cboMsgList.ItemData

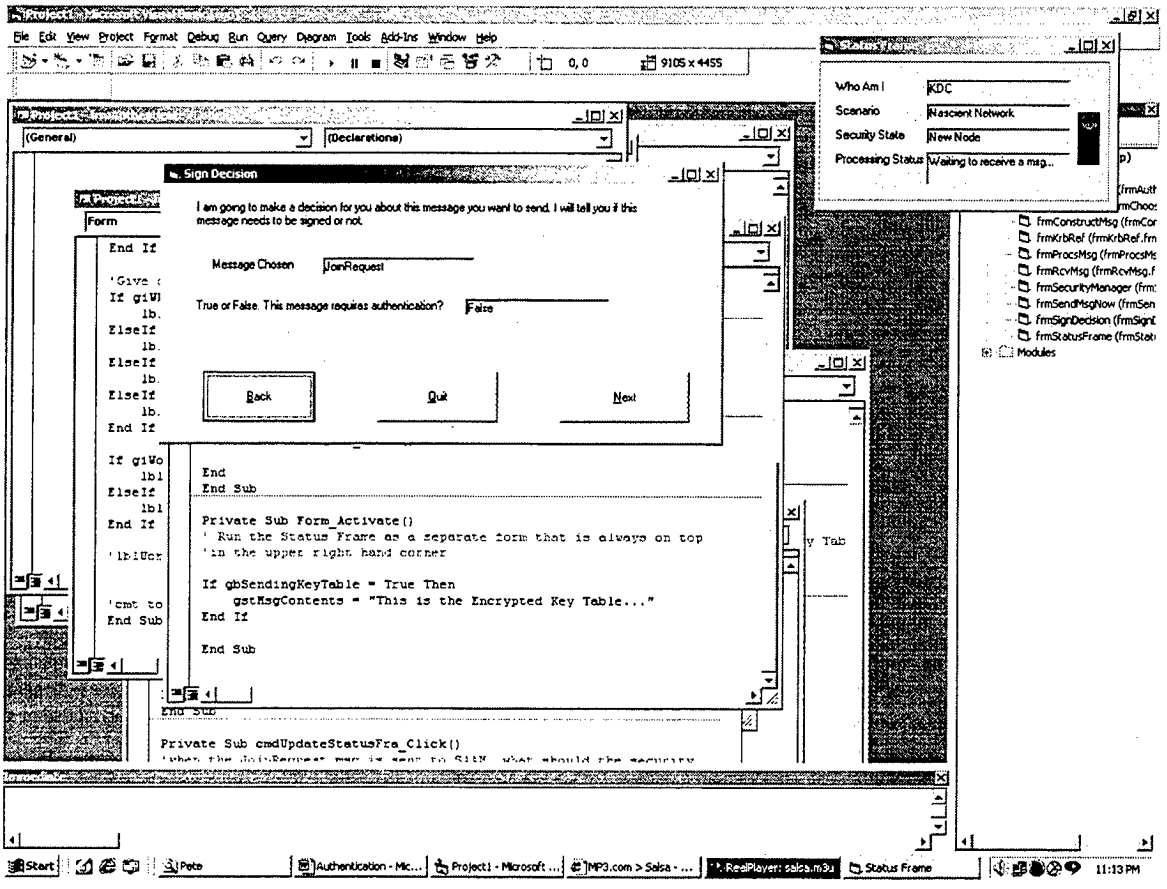
'If cboMsgList.Text = "JoinRequest" Then

' giWorkingMsg = 1

```
'End If

frmStatusFrame.Hide

frmStatusFrame.Show vbModeless

End Sub

Private Sub cmdBack_Click()

Me.Hide

'force a refresh of the status frame

frmStatusFrame.Cls

frmSecurityManager.Hide

frmSecurityManager.Show vbModal

End Sub

Private Sub cmdNext_Click()

Me.Hide

frmSignDecision.Show vbModal

End Sub
```

```
Private Sub cmdQuit_Click()


End

End Sub


Private Sub Form_Activate()

'Run the Status Frame as a separate form that is always on top

'in the upper right hand corner


If gbSendingKeyTable = True Then

   gstMsgContents = "This is the Encrypted Key Table..."

End If


End Sub
```

D. FRMSIGNDECISION



'Dim mbSignDecision As Boolean 'stores sign or don't sign this message

Private Sub cmdBack_Click()

Me.Hide

frmChooseMsg.Hide

frmChooseMsg.Show vbModal

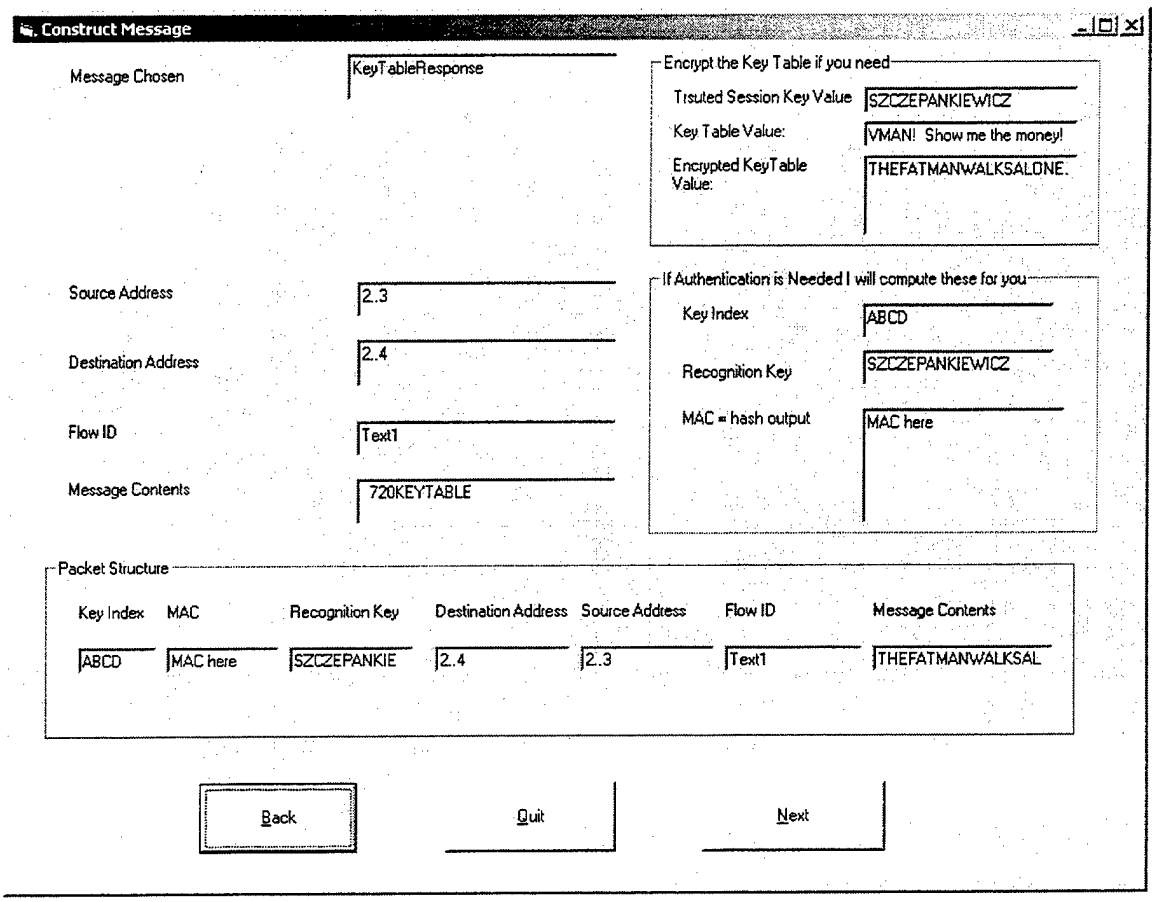
End Sub

Private Sub cmdNext_Click()

```
Me.Hide
frmConstructMsg.Show vbModal
End Sub

Private Sub cmdQuit_Click()

End
End Sub

Private Sub Form_Activate()
'get the name of the message chosen to send and display it.

lblMsgChosen.Caption = frmChooseMsg.cboMsgList.Text
If gbSendingKeyTable = True Then
   'gstMsgContents = "This is the Encrypted Key Table..."
   lblMsgChosen.Caption = "KeyTableResponse"
End If

'Make sign decision.  Knowledge Engine.
mbSignDecision = True 'Default to require signing.
If frmChooseMsg.cboMsgList.Text = "JoinRequest" Then
' mbSignDecison = False
```

```
    lblSignDecision.Caption = "False" 'Display signing decision results
    ElseIf lblMsgChosen.Caption = "KeyTableResponse" Then
'   mbSignDecison = False
    lblSignDecision.Caption = "False" 'Display signing decision results
    'Add all the msgs that do NOT require signing here.
End If

If frmChooseMsg.cboMsgList.Text = "TrustedSessionRequest" Then
    lblSignDecision.Caption = "True" 'Display signing decision results
End If

End Sub
```

E. FRMCONSTRUCTMESSAGE




```
Dim mstSourceAddr   As String

Dim mstDestAddr     As String

Dim mstMsgContents  As String


Private Sub cmdBack_Click()


Me.Hide

 frmSignDecision.Show vbModal

End Sub
```

```
Private Sub cmdNext_Click()


Me.Hide

frmSendMsgNow.Show vbModal

End Sub


Private Sub cmdQuit_Click()


End

End Sub


Private Sub Form_Activate()


If frmSignDecision.lblMsgChosen = "JoinRequest" Then

   txtMsgType.Text = "JoinRequest"

   mstSourceAddr = "1..2"

   txtSrcAddr.Text = mstSourceAddr

   mstDestAddr = "1..1"

   txtDestAddr.Text = mstDestAddr

   mstMsgContents = "  LETMEJOIN "

   txtMsgContents.Text = mstMsgContents

   lblFlowID.Visible = False
```

```
    txtFlowID.Visible = False
    lblKeyIndex.Visible = False
    txtKeyIndex.Visible = False
    lblMsgAC.Visible = False
    txtMsgAC.Visible = False
    lblRecognitionKey.Visible = False
    txtRecognitionKey.Visible = False
    lblFloID.Visible = False
    txtFloID.Visible = False
    fraEncryptKeyTbl.Visible = False
    fraIfAuthNeeded.Visible = False
    txtMessageContents.Text = mstMsgContents
End If

If frmSignDecision.lblMsgChosen = "TrustedSessionRequest" Then
    txtMsgType.Text = "TrustedSessionRequest"
    mstSourceAddr = "131.120.8.155"
    txtSrcAddr.Text = mstSourceAddr
    mstDestAddr = "131.120.9.66"
    txtDestAddr.Text = mstDestAddr
    mstMsgContents = "  TRUSTME "
    txtMsgContents.Text = mstMsgContents
    lblFlowID.Visible = True
```

```
    txtFlowID.Visible = True
    lblKeyIndex.Visible = True
    txtKeyIndex.Visible = True
    lblMsgAC.Visible = True
    txtMsgAC.Visible = True
    lblRecognitionKey.Visible = True
    txtRecognitionKey.Visible = True
    lblFloID.Visible = True
    txtFloID.Visible = True
    fraEncryptKeyTbl.Visible = False
    fraIfAuthNeeded.Visible = True
    txtKeyIndex.Text = txtKI.Text
    txtMsgAC.Text = txtMAC.Text
    txtRecognitionKey.Text = txtRecKey.Text
    txtFloID.Text = txtFlowID.Text
    txtMessageContents.Text = mstMsgContents
End If

If frmSignDecision.lblMsgChosen = "KeyTableResponse" Then
    txtMsgType.Text = "KeyTableResponse"
    mstSourceAddr = "2..3"
    txtSrcAddr.Text = mstSourceAddr
    mstDestAddr = "2..4"
```

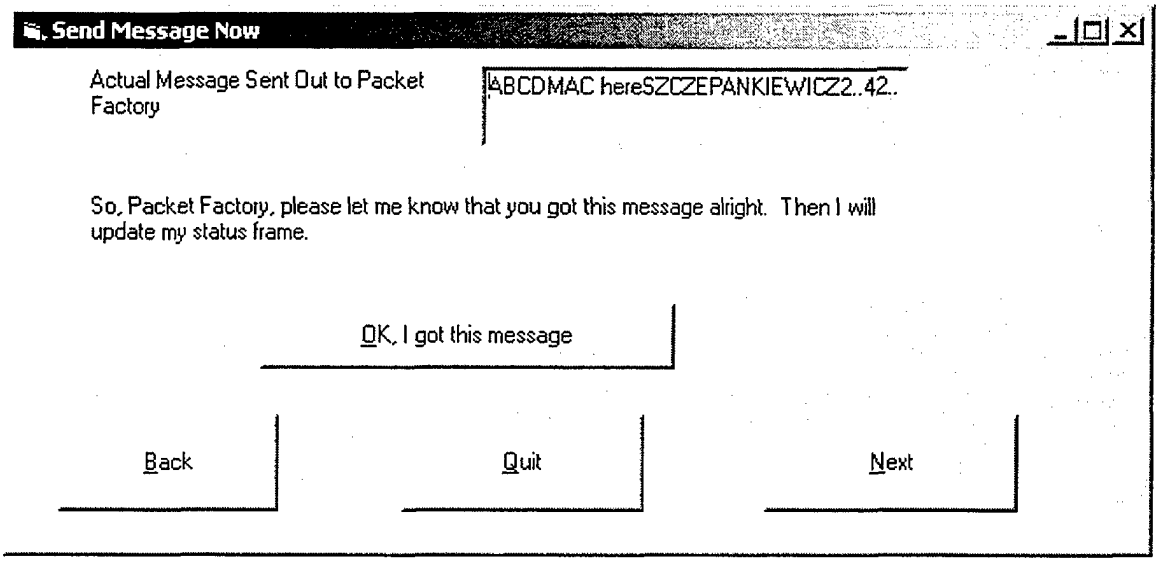
```
    txtDestAddr.Text = mstDestAddr
    mstMsgContents = "  720KEYTABLE "
    txtMsgContents.Text = mstMsgContents
    lblFlowID.Visible = True
    txtFlowID.Visible = True
    lblKeyIndex.Visible = True
    txtKeyIndex.Visible = True
    lblMsgAC.Visible = True
    txtMsgAC.Visible = True
    lblRecognitionKey.Visible = True
    txtRecognitionKey.Visible = True
    lblFloID.Visible = True
    txtFloID.Visible = True
    fraEncryptKeyTbl.Visible = True
    fraIfAuthNeeded.Visible = True
    txtKeyIndex.Text = txtKI.Text
    txtMsgAC.Text = txtMAC.Text
    txtRecognitionKey.Text = txtRecKey.Text
    txtFloID.Text = txtFlowID.Text
    txtMessageContents.Text = txtEncryptedKeyTable.Text
End If


'Build Packet structure before passing back to SAAM.
```

txtDestinationAddr.Text = mstDestAddr

txtSourceAddr.Text = mstSourceAddr

End Sub

F. FRMSENDMSGNOW



Private Sub cmdBack_Click()

Me.Hide

frmConstructMsg.Show vbModal

End Sub

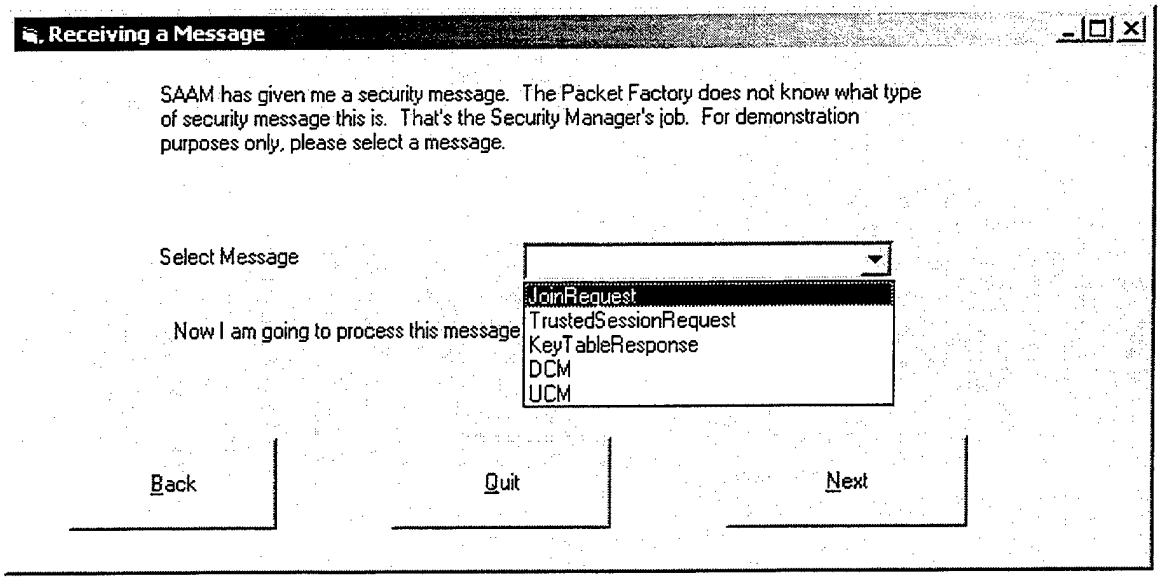
Private Sub cmdNext_Click()

Me.Hide

```
'frmSecurityManager.Hide

Me.Hide
frmSecurityManager.Show vbModal
End Sub

Private Sub cmdQuit_Click()

End
End Sub

Private Sub cmdUpdateStatusFra_Click()
'when the JoinRequest msg is sent to SAAM, what should the security
'Manager do next?
'I am still in the same role: RouterB
'I am still a New Node, not yet Trusted.
'I am still in the new join scenario
'The only thing to change is my processing state.
'Now I am waiting for a kerberos ticket and authenticator from
'RouterA.
frmStatusFrame.lblProcsStatus_Val.Caption = "Waiting for a Ticket +
Authenticator..."
```

```
End Sub

Private Sub Form_Activate()

gstMsgContents = frmConstructMsg.txtKeyIndex.Text & _
                 frmConstructMsg.txtMsgAC.Text & _
                 frmConstructMsg.txtRecognitionKey.Text & _
                 frmConstructMsg.txtDestinationAddr.Text & _
                 frmConstructMsg.txtSourceAddr.Text & _
                 frmConstructMsg.txtFloID.Text & _
                 frmConstructMsg.txtMessageContents.Text

If gbSendingKeyTable = True Then
   gstMsgContents = "This is the Encrypted Key Table..."
   gbSendingKeyTable = False
End If

txtMsgSent = gstMsgContents

End Sub
```

G. FRMRCVMSG

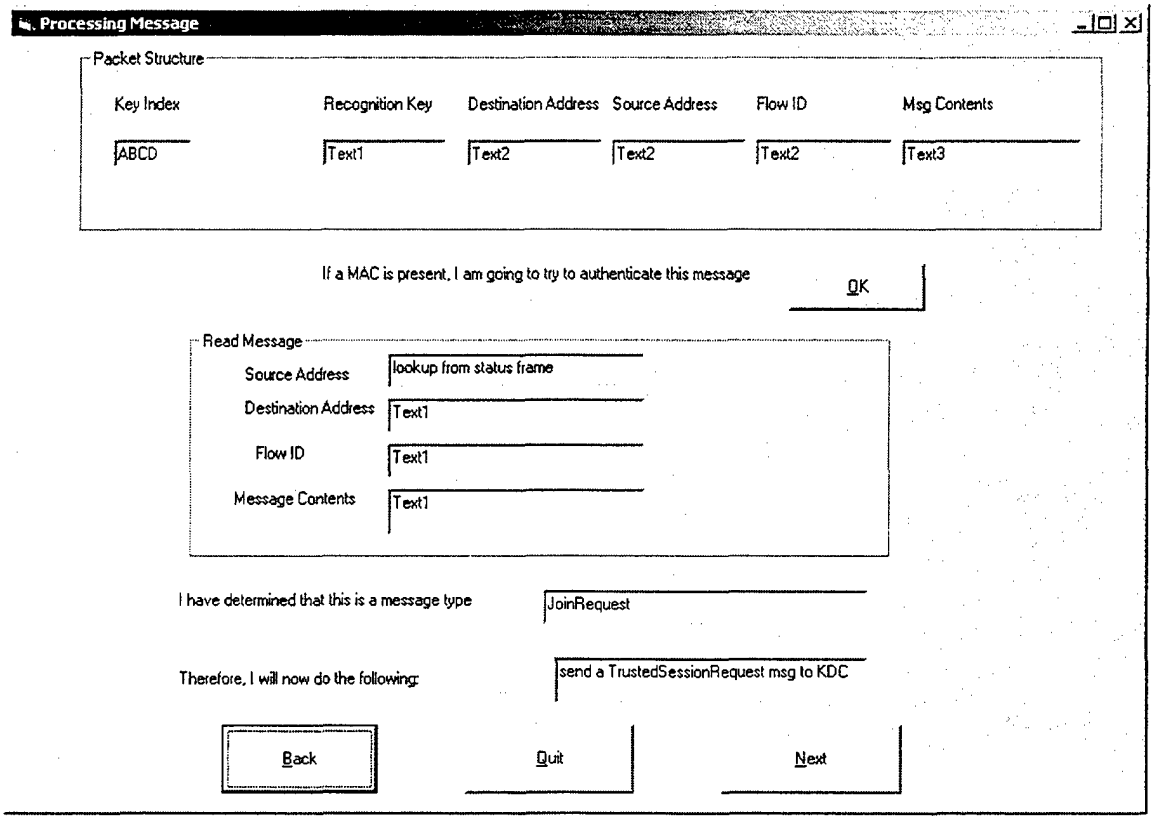



```
Private Sub cmdBack_Click()


Me.Hide

frmSecurityManager.Show vbModal

End Sub


Private Sub cmdNext_Click()


Me.Hide

frmProcsMsg.Show vbModal

End Sub


Private Sub cmdQuit_Click()
```

End

End Sub

H. FRMPROCSMSG



Private Sub cmdBack_Click()

Me.Hide

frmRcvMsg.Show vbModal

End Sub

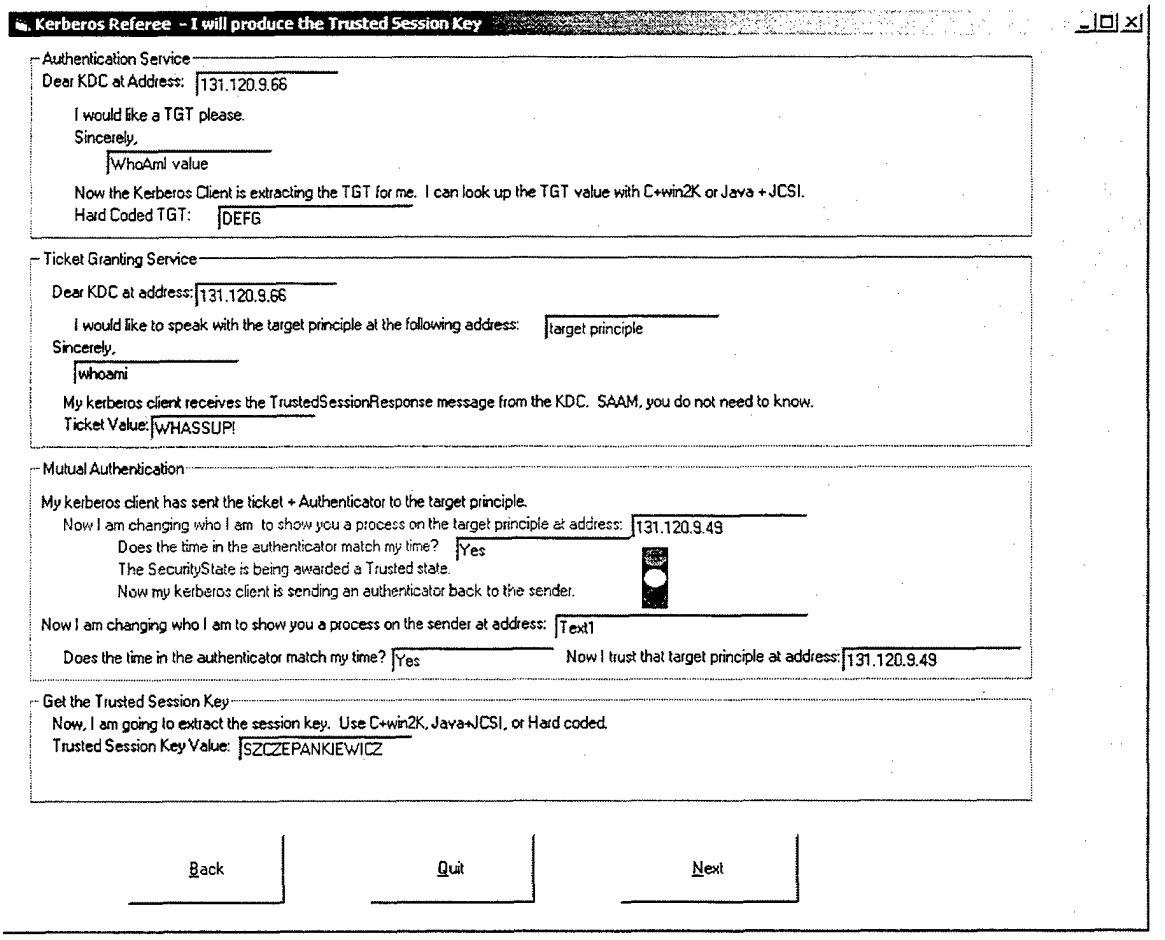
Private Sub cmdNext_Click()

Test where to go next.

```
'If There is Kerberos stuff to do, then go to the KerberosReferee.

Me.Hide
frmKrbRef.Show vbModal

'If there is regular securityManager stuff to do, then go to the Security Manager.

Me.Hide
frmSecurityManager.Show vbModal
End Sub

Private Sub cmdQuit_Click()

End
End Sub

Private Sub Form_Activate()

'case KeyTableResponse, don't show mac because it is not signed.
'lblMsgChosen.Caption = frmChooseMsg.cboMsgList.Text
'If frmRcvMsg.cboMsgList.ItemData = KeyTableResponse Then
  lblMsgAC.Visible = False
  txtMsgAC.Visible = False
```

End Sub

I. FRMKRBREF



Private Sub cmdBack_Click()

Me.Hide

frmProcsMsg.Show vbModal

```
End Sub


Private Sub cmdNext_Click()


gbSendingKeyTable = True 'Remember that I am going to send the Key Table now.
Me.Hide
frmSignDecision.Show vbModal
End Sub


Private Sub cmdQuit_Click()
End
End Sub
```

LIST OF REFRENCES

1. Aberdeen Group. Legato Continuum: Delivering "Information Continuance." www.aberdeen.com/cgi-bin/rf.cgi?doc_id=08991539

2. Ahmad. "Windows Time Synchronization Service" Windows 2000 Magazine. March 2000.

3. Akkoc. Autoconfiguration in SAAM. Thesis, Naval Post Graduate School. June 2000.

4. Baker, Lindell, and Talwar. RFC2747. RSVP Cryptographic Authentication. January 2000.

5. Burton and Obel. Strategic Organizational Diagnosis and Design: Developing Theory for Application 1998.

6. Deitel and Deitel. Java How To Program 3rd Ed. Prentice Hall. 1999.

7. Greenberg. Managing Behavior in Organizations, Prentice Hall. 1999.

8. Hensley and Ludden. "ATM Security via "Stargate" Solution. Thesis, Naval Post Graduate School. September 1999.

9. Joint Chiefs of Staff. Joint Vision 2010.

10. Joint Chiefs of Staff. Joint Pub 3-13 Joint Doctrine for Information Operations. 1998.

11. Kerr. "On the Folly of Rewarding A, While Hoping for B" Academy of Management Journal. No. 4:769-83. 1975.

12. Kohl. RFC1510. The Kerberos Network Authentication Service (V5). September 1993.

13. Lewis, Ted. Microsoft Rising and Other Silicon Valley Tales. IEEE Computer Society Publications. 1999.

14. Linn. RFC1964. The Kerberos Version 5 GSS-API Mechanism. June 1996.

15. Microsoft White paper . The Security Support Provider Interface (SSPI). February 1999.

16. Mills. rfc1769 - Simple Network Time Protocol. March 1995.

17. Quek, Henry. QoS Management With Adaptive Routing for Next Generation Internet. Thesis, Naval Post Graduate School. March 2000.

18. Redshift. T-1 price quotes corp.redshift.com

19. Stallings, William. Cryptography and Network Security 2nd ed. Prentice Hall. 1998.

20. Tung, Brian. Kerberos: A Network Authentication System. Addison-Wesley Pub. Co. May 1999.

21. Vrable and Yarger. The SAAM Architecture: Enabling Integrated Services. Thesis, Naval Post Graduate School. September 1999.

22. Xacct Corporation www.xacct.com

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center.....2
   8725 John J. Kingman Road, Ste 0944
   Ft. Belvoir, Virginia 22060-6218

2. Dudley Knox Library.....2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, California 93943-5101

3. Director, Training and Education.....1
   MCCDC, Code C46
   1019 Elliot Road
   Quantico, VA 22134-5027

4. Chairman, Code CS.....1
   Computer Science Department
   Naval Postgraduate School
   Monterey, CA 93940-5000

5. Director, Marine Corps Research Center.....2
   MCCDC, Code C40RC
   2040 Broadway Street
   Quantico, VA 22134-5107

6. Marine Corps Representative.....1
   Naval Post Graduate School
   Code 037, Bldg 330, Ingersoll Hall, Room 116
   555 Dyer Road
   Monterey, CA 93943

7. Marine Corps Tactical Systems Support Activity.....1
   Technical Advisory Branch
   Box 555171
   Camp Pendleton, CA 92055-5080

8. Dr. Geoffrey Xie.....1
   Computer Science Department, Code CS
   Naval Postgraduate School
   Monterey, California 93943-5100

9. Mr. Rex Buddenberg.....1
Information Systems Department, Code IS/BU
Naval Postgraduate School
Monterey, California 93943-5100

10. Mr. Cary Colwell.....1
Computer Science Department, Code CS
Naval Postgraduate School
Monterey, California 93943-5100

11. LT Peter Szczepankiewicz 1
Fleet Information Warfare Center
2555 Amphibious Drive
Norfolk, VA 23521

12. Capt. Luis Velazquez.....1
1828 Aberdeen Circle
Crofton, MD 21114

13. Dr. Bill Richter.....1
SPAWARSTSCEN
P. O. Box 190022
North Charleston, SC 29419-9022

14. Mr. and Mrs. Gary Szczepankiewicz.....1
319 Dushane Dr.
Buffalo, NY 14223